

INTELLIGENT DATA UNDERSTANDING FOR ARCHITECTURE ANALYSIS OF ENTRY, DESCENT, AND LANDING: APPLICATIONS FOR MARS MISSIONS

A Thesis

Presented to the Faculty of the Graduate School
of Cornell University

in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Aerospace Engineering

by

Samalis Santini de Leon

August 2018

© 2018 Samalis Santini De Leon
ALL RIGHTS RESERVED

ABSTRACT

Because Entry, Descent and Landing (EDL) system validations are limited in Earth environments, these technologies rely heavily on modeling and analysis tools to evaluate system performance and capture uncertainties, which determine the success of a mission. However, the current approach suffers from an important limitation. While the subject matter expert can leverage his or her knowledge and expertise with past systems to identify areas of risk and features of interest in the datasets available, the next generation of EDL systems may present unprecedented challenges that may be missed by the human. Landing humans on Mars, for example, will pose unprecedented challenges to EDL technologies driven by the need to land larger payload mass with extremely high reliability and safety requirements. The goal of this research is to advance the state of the art for offline Intelligent Data Understanding (IDU) technologies for Entry, Descent and Landing (EDL) by incorporating an intelligent assistant - called Daphne/EDL - that supports humans in architecting problems specific to the field of EDL. In this thesis, we describe a first prototype of the Daphne/EDL assistant in the context of three use cases to cover a range of representative problems in EDL architecture analysis and to show the capabilities of the assistant to support those use cases. Specifically, we demonstrate the baseline functionalities of the EDL assistant that include: 1) preliminary analysis capabilities of simulation datasets; 2) extraction of performance metrics from a historical database; and 3) automated generation of a scorecard. The scorecard contains metrics critical to assess EDL architecture performance. The assistant highlights metrics that fall out of spec of a design and communicates them to

the subject-matter expert.

BIOGRAPHICAL SKETCH

Samalis Santini De Leon completed a B.S. in Mechanical Engineering from the University of Puerto Rico at Mayaguez on May 2016. During her freshman year she was a recipient of the Robert C. Byrd Fellowship for academic excellence. For the following years she became a NASA Space Grant fellow and became one of the team leads for the RockSat-X Project with the Colorado Space Grant Consortium and the Wallops Flight Facilities. During these years she was an also an Aeromechanics intern at the NASA Ames Research Center, an undergraduate researcher at the Massachusetts Institute of Technology (MIT), and a Co-op at the MITRE Corporation with the Mechanical Engineering and Reliability Branch.

At the moment, she is completing her second year of graduate studies and will graduate with a Masters of Science in Aerospace Engineering from the Mechanical and Aerospace Engineering Department at Cornell. Samalis works under the supervision of Professor Daniel Selva and her focus is architecture analysis for entry, descent, and landing on Mars, specifically. Her research has been supported by the NASA Science and Technology Research Fellowship (NSTRF).

During the following years she will continue her pursuit of a PhD degree at Texas A& M University under the continued supervision of Professor Selva and support from NSTRF. For the remainder of her graduate studies she will focus on expanding the capabilities of intelligent data understanding (IDU) technologies for architecture analysis of entry, descent, and landing.

ACKNOWLEDGEMENTS

First, I would like to thank my advisor Dr. Daniel Selva for his time and patience. Working with him and the students at the SEAK Lab has given me exposure to the field of system architecture. All of the feedback and advice from him has been helpful for me to grow as a professional. I am also thankful for the time Prof. Selva has taken to advise me as his graduate student and for all of the social events put together for the lab. The SEAK lab has been like a second family and I am very thankful for the support from everyone.

I would also like to thank Professor Mason Peck for introducing me to the Aerospace Engineering Department at Cornell. My career path was very uncertain at the moment I met Dr. Peck and perhaps I wouldn't have ended up in such an exceptional institution if we had not met and I learned about everything Cornell has to offer. I am especially thankful for the opportunity. I would like to thank Marcia Sawyer for everything she does for us the grad students to make us feel happy in the department.

To Dr. William Warmbrodt at the NASA Ames Research Center all I can say is: "How cool is that?" One milestone down and many more to go.

This research was funded by the NASA Science Research and Technology Fellowship (NSTRF) and thanks to this opportunity I have had the opportunity to collaborate with Dr. David Way. I am especially thankful for his patience and the time he has taken to expose me to the field of Entry, Descent, and Landing and all of his perspective as a researcher in EDL on how my research can contribute to the EDL teams. Dr. Way incorporated me into the EDL team at Langley and JPL from day one and I am very thankful for the opportunity to work on an ongoing mission to Mars.

I would like to thank my two special groups of friends. First, the "Pretty

Ladies Coven” has supported me through my endeavors. We have been friends since we were only 12 years old and we still stay in touch every day. Every single one of these women is a fighter and each went through their own set of obstacles and now we have doctors, scientists, physicists, and adventurers. I admire you all greatly. Shout out Kritzia Pabón, Marialis Rosario, Keysha Pietri, Lyann Martínez, Selena Rodríguez, Marlys Massini, and Paloma Rodríguez. And as we go through rough times, let’s not forget what the esteemed Benito Antonio Martínez and any other puertorican proverb would say: “No te preocupes, tamos bien”

Second, I want to thank the family of “Maya” friends I got during my undergraduate studies. I never thought I’d make lifelong friends in a single place. The most memorable moments I have been through has been with you guys. Thank you Luisito, Coral, Manda, Lizama, Escobar, Calderon, Jongui, Marco, Petrus, Ricky, Gaco, Brenda, Marjo, Hector, Eti, Pirulo, Yolanda, and Ileana for your continued support up to the present day. Shout out to all of you and as you all tell me (yes, another terrible Puertorican proverb): “El miedo lo dejamos en la gaveta.”

Last but not least, I would like to thank my family. Especially my parents. My mom and dad have been through more than anyone I know. My parents are a true source of inspiration and I am eternally grateful for all of the support they have given me from day one. Science was always a passion of mine since I was little and my biggest fear was not being able to obtain an advanced degree due to the hardships my family went through. Nevertheless, my parents both reminded me that I can achieve whatever I propose myself. This encouragement helped me get through rough times and made me the person I am today. I would like to thank my cousin Vanessa and her wife Lynnette for their uncon-

ditional support. Both of you are like my second parents and you are both truly admirable. My family is too big to enumerate so I will proceed to thank all of the Santini family.

TABLE OF CONTENTS

Biographical Sketch	iii
Acknowledgements	iv
Table of Contents	vii
List of Tables	ix
List of Figures	x
1 Introduction	1
1.1 Motivation	1
1.2 Thesis Objectives	3
1.3 Thesis Outline	6
2 Entry, Descent, and Landing	8
2.1 Mars Entry Descent and Landing Overview	8
2.1.1 Past Successful Mars Missions EDL Architecture	10
2.1.2 Equations of Motion for Planetary Entry	11
2.1.3 Aeroheating During Planetary Entry	18
2.2 Limitations of EDL Technologies	18
3 Simulations for Entry, Descent and Landing	21
3.1 POST-II-based Monte Carlo Dispersion Analysis	22
3.2 Sensitivity Analysis of POST-II Based Monte Carlo Dispersion Analysis	26
4 Intelligent Data Understanding	27
4.1 Intelligent Data Understanding (IDU) Technologies	27
4.1.1 State of the Art of IDU Technologies	28
4.2 Cognitive Assistants for Space Applications	30
4.3 Daphne Architecture	33
4.4 Daphne for EDL	34
4.4.1 Frontend	35
4.4.2 Daphne Brain	35
4.4.3 Backends	40
4.4.4 Data Sources	41
5 Case Study	45
5.1 Use Case 1: Rapid Assessment of of Mission-Specific System Per- formance	45
5.2 Use Case 2: Accessing and Analyzing Simulation Dataset	47
5.2.1 Context and Goals of the Simulation	47
5.2.2 Simulation Setup	48
5.2.3 Skipout Angle Due to Dispersion Entry Velocity and EFPA	49
5.2.4 Results	50
5.2.5 Interacting With Daphne	52

5.3	Use Case 3: Comparing Simulation Results to Nominal Values of Previous Landed Missions	58
6	Conclusions	62
	Bibliography	64

LIST OF TABLES

2.1	Summary of EDL architecture of previous Mars missions [1]. . .	12
2.2	Variable definition for the equations of planar motion during ballistic flight.	15
5.1	Entry States and Heat Shield Parameter for Simulation	49
5.2	Cases evaluated for Monte Carlo simulation to measure the Effect of uncertainty in entry velocity and EFPA to skipout angle .	50
5.3	Simulation results for skipout angle due to sensitivities to entry velocity and ballistic coefficient.	56

LIST OF FIGURES

2.1	Mars Science Laboratory (MSL) EDL sequence.	8
2.2	Schockwave separation for the models created for the Mercury re-entry vehicle concepts.	10
2.3	Planar flight FBD [12]	14
3.1	POSTII simulation models and data flow [4].	22
3.2	Current models evaluated in POSTII	23
3.3	Current models evaluated in POSTII	25
4.1	Daphne Brain Architecture	34
4.2	Question classification in Daphne.	36
4.3	Envisioned question types for Daphne.	37
4.4	Query template for historical database.	38
4.5	EDL database schema.	42
5.1	Follow-up questions for Daphne (Historical Database)	46
5.2	Skipout angle variation due to uncertainty in entry velocity and EFPA	51
5.3	Steepest Skipout angle variation due to entry velocity	52
5.4	Use Case	54
5.5	Follow-up questions for Daphne	55
5.6	Follow-up questions for Daphne	57
5.7	Deceleration for three random cases in which lofting occurs for an entry vehicle of $\beta = 300kg/m^2$	58
5.8	Velocity as a function of altitude for three random cases in which lofting occurs for an entry vehicle of $\beta = 300kg/m^2$	59
5.9	Follow-up questions for Daphne (Historical Database)	61

CHAPTER 1

INTRODUCTION

1.1 Motivation

Planetary entry, descent and landing (EDL) is one of the most challenging phases in interplanetary missions. For landing on Mars, atmospheric entry becomes increasingly challenging due to the fact that the Martian atmosphere is roughly a hundred times less dense than Earth's. Therefore, EDL systems must be capable of decelerating from hypersonic to subsonic speeds at low altitudes, thus reducing the time for subsequent events in the EDL sequence to occur [1]. Furthermore, during EDL vehicles experience high aerothermal loading due to friction with the atmosphere. During this phase, roughly 95% of the energy is dissipated through thermal heating and entry vehicles experience peak heat rates up to 200 W/cm^2 (e.g. Mars Science Laboratory). Aerodynamic forces acting on the entry vehicle can generate peak deceleration loads over 10 Earth G's. Whereas robotic missions can withstand such g-loads, future manned missions to Mars will require minimizing the peak deceleration (5-Gs or less) and peak heat load of the entry vehicle due to the presence of humans. These stringent requirements for manned missions will require new technologies and entry configurations to ensure the safety of the crew.

Currently, fundamental EDL technologies used for robotic landings on Mars have been derived and extended from the capabilities developed for Apollo during the 1960s and 1970s. The current state of the art for EDL architecture analysis is defined by the Mars Science Laboratory (MSL) EDL architecture. This architecture consisted of six segments: Exo-atmospheric flight, Guided Entry,

Parachute Descent, Powered Descent, Sky Crane, and Flyaway. Many elements of the MSL architecture elements were derived and extended from the Viking, Pathfinder (MPF) and the Mars Exploration Rover (MER) missions, resulting in improved target precision and successful landing [2]. However, the next generation of MSL-class landing vehicles will be delivering higher payloads at tighter delivery ellipses that are pushing the limits of current EDL technologies [3]. Landing humans on Mars (or Europa for example) will also pose unprecedented challenges to EDL technologies driven by the need to land a larger payload mass with extremely high reliability and safety requirements.

In order to design the next generation of EDL systems, it is necessary to analyze a wide range of architectures with high fidelity simulations of performance, cost, schedule and risk that take into account uncertainty. Because Earth-based testing of Mars environmental conditions is limited, performance analysis of EDL systems under uncertainty relies on high-fidelity engineering end-to-end simulation. EDL simulations provide a platform to support system design, trade studies, development, testing, and operations of a given mission. For previous Mars missions, such as MSL, the NASA EDL team used the Program to Optimize Simulated Trajectories II (POST2) to simulate a variety of entry conditions under many model parameterizations (e.g. gravity, planetary geometry, atmospheric, aerodynamic, control system, guidance, and navigation models) and evaluate performance, mission-level feasibility, design trades, among other capabilities [4].

Assessment and success criteria evaluation of a proposed mission/system architecture is achieved by using Monte Carlo Analysis. To quantify the robustness of an entry vehicle system requires executing Monte Carlo simulations for

varying configurations (e.g. lift-to-drag ratio, entry flight path angle) as well as varying atmospheric conditions, among others. Inevitably, this results in very large data sets. For the MSL mission, for example, simulation runs were in the order of thousands, resulting in millions of individual cases [5]. These large datasets generated are manually analyzed by the subject matter expert, trying to find interesting correlations and couplings between parameters, and assess the sensitivity of figures of merit to various parameters. This analysis work is important since it may lead to the discovery of a major flaw in a design, for example. However, the current approach suffers from an important limitation. While the subject matter expert can leverage his or her knowledge and expertise with past systems to identify issues and features of interest in the dataset, the next generation of EDL systems may present unprecedented challenges that may be missed by the human. This expert-based analysis also neglects to reflect the effect of these uncertainties in the overall system architecture of the mission. Because of this limitation, there is a need for an end-to-end architecture analysis. Such analysis is critical for determining mission feasibility and for identifying areas of risk that drive architectural the decision making process throughout a mission's lifecycle.

1.2 Thesis Objectives

The objective of this thesis is to describe the baseline functionalities of a cognitive assistant (CA) for advancing the state of the art of offline Intelligent Data Understanding Technologies (IDU) for architecture analysis of EDL. The rationales for exploring the role of Intelligent Data Understanding technology for performance analysis of EDL systems are: 1) the concept of knowledge engi-

neering has been applied for operational processes to define the knowledge involved in the given process but its use in performance analysis has not been established. 2) the use of an intelligent system during data analysis can help the subject matter expert identify problems with a design earlier, which may lead to less rework, large cost savings, and increased probability of success.

It is envisioned that the proposed intelligent assistant will provide a platform for engineers to evaluate EDL architecture trades and technology for architecture development decisions and verify EDL mission performance while considering uncertainties to ensure mission success. In this thesis emphasis is given to using the Daphne/EDL assistant for automating several steps in the EDL data analysis process such as data extraction from multiple sources (e.g. simulation results, database). Capabilities developed during this thesis can be adapted to establish baseline functionalities of a cognitive assistant for crew support in space. Astronauts undergo decision-making tasks in the face of uncertainty or off-nominal situations. The work developed in this thesis can be adapted to serve as a cognitive assistant for spacecrew support. The Daphne/EDL assistant satisfies preliminary functional design requirements outlined in Reference [27] for data extraction of data collected and can provide information during the data analysis process. Furthermore, the cognitive assistant can make use of the database for accessing guidelines for off-nominal cases and the respective actions that need to be taken. The astronauts can ask about a particular scenario using natural language and can obtain relevant information about the situation.

To demonstrate the preliminary capabilities of the EDL assistant, we present a case study of a ballistic atmospheric entry simulation. Daphne's EDL analysis capabilities however, can be employed for analysis of a wide variety of EDL

architectures. A ballistic entry simulation was selected due to its simplicity and given that it also presents high cost savings relative to entry vehicle systems with active attitude control systems. In the case presented, we are interested evaluating the effect of uncertainty in entry velocity to atmospheric skip-out angle at shallow entry flight path angles. Here, emphasis is given to Daphne's data extraction capabilities from various sources of knowledge as well as written and verbal communications, which are two key aspects of an intelligent assistant for data analysis. Using multiple sources of knowledge is useful for comparing performance metrics and for performing trade studies in the early stages of a mission lifecycle: Pre-Phase A (Concept Studies) to Phase A (Concept and Technology Development).

It must be noted that the work performed for this thesis focused on establishing the baseline functionalities for data extraction from simulation results and two sources of knowledge: a historical database and a scorecard. These functionalities pave the way for future work given that there is no record of a cognitive assistant employed for EDL architecture performance. To take advantage of the potential of incorporating a cognitive assistant for high-fidelity architecture analysis, future work includes developing algorithms for data mining of EDL datasets, for example, to obtain insight on a particular design. Capabilities of the intelligent assistant will be further enhanced by having the system communicate critical issues or provide "critiques" based on information contained in the sources of knowledge. To achieve a mixed-initiative approach, both the subject matter expert and the assistant shall provide their expertise and ask questions as needed to achieve common goals, such as improving the data mining process.

1.3 Thesis Outline

Chapter 2, Section 2.1 provides an overview of EDL on Mars. Section 2.1.1 provides a summary of past landed missions on Mars. Section 2.1.2 introduces the fundamental equations of motion for atmospheric entry. This section will discuss the equations of planar motion and the analytical models for atmospheric entry (i.e. Allen-Eggers approximations). Section 2.1.3 provides an overview of aeroheating and the Sutton-Graves equations for stagnation point convective heat rate estimation. Finally, section 2.2 discusses the limitations of current EDL technologies.

Chapter 3 provides an overview of the NASA Langley Research Center (LARC) Program to Optimize Simulated Trajectories II (POST-II), a high-fidelity engineering simulation employed by NASA to support EDL system design throughout all phases of a mission’s lifecycle. Subsection 3.1 will discuss Monte Carlo dispersion analysis used to quantify robustness and risk associated with a design.

Chapter 4 presents an overview of the state of the art in IDU technologies and intelligent assistants technologies. In this Chapter we also introduce Daphne, a cognitive assistant for analysis of EDL datasets. Some of its preliminary capabilities are discussed.

Chapter 5 presents a case study for a ballistic entry mission subjected to varying entry conditions. In this chapter, emphasis is given to the effect of dispersed atmospheric density during planetary entry at shallow entry flight-path angles. Section 5.2.1 provides an overview of the context and objectives of the case study. Section 5.2.2 gives an overview of the models used in the POST-

II-based Monte Carlo simulation. Finally, Section 5.2.4 will discuss the results of the case study and will also demonstrate Daphne's current capabilities with respect to EDL.

Finally, Chapter 6 summarizes the findings and contributions of the thesis. Chapter 6 highlights limitations of the analysis and emphasizes the opportunities for Daphne to conduct sensitivity analysis on EDL datasets.

CHAPTER 2

ENTRY, DESCENT, AND LANDING

2.1 Mars Entry Descent and Landing Overview

Planetary EDL consists of a series of events and maneuvers required to land a payload, or vehicle, on the surface of a planet. On Mars, the EDL phase lasts roughly 7 minutes for a 899 kg payload, given that the the planet's atmosphere is roughly $1/100^{th}$ of Earth's density, making deceleration at higher altitudes challenging.

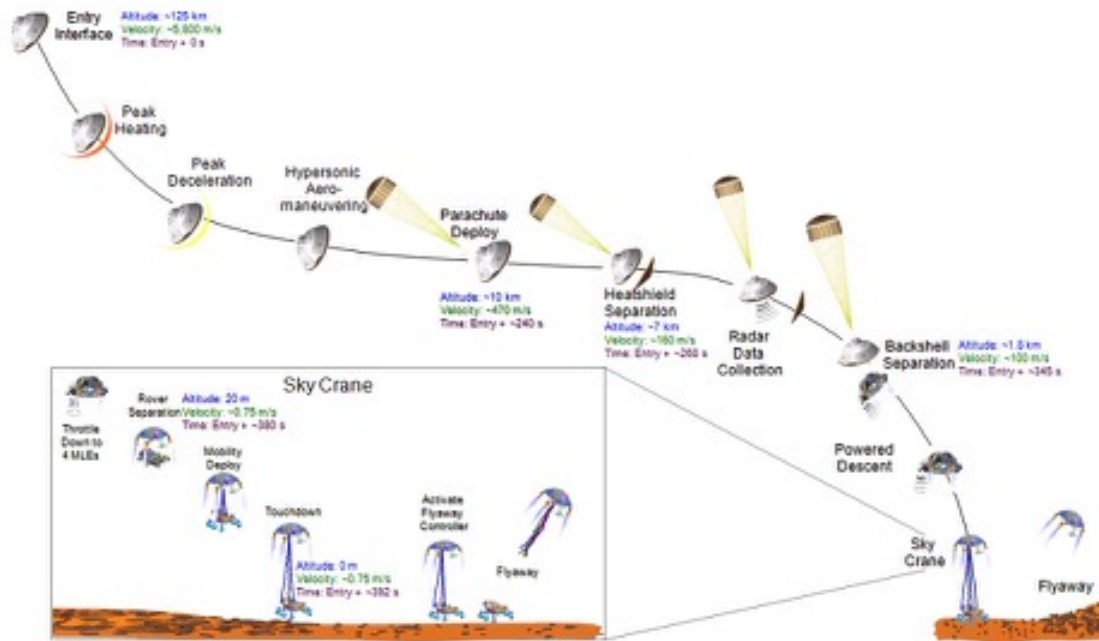


Figure 2.1: Mars Science Laboratory (MSL) EDL sequence.

Figure 2.1 depicts the state-of-the-art EDL architecture, defined by MSL. This mission incorporated Viking, Mars Pathfinder (MPF) and Mars Exploration Rovers (MER) heritage technologies and extended the landed mass capabilities for landing on Mars. On August 5, 2012, following cruise stage separation, MSL entered the Martian atmosphere performing the first guided entry at an altitude of about 125 km above the surface at a planet-relative velocity of approximately 5.8 km/s, and an entry flight path angle (EFPA) of approximately -15° [6]. Not only was MSL the first mission that featured guided entry on Mars, but it also became the heaviest payload landed successfully on the surface of Mars.

The entry segment is one of the most critical events in the EDL sequence. As seen in Figure 2.1, during this segment the entry vehicle undergoes peak heating (for MSL the heat shield reached $2,100^\circ\text{C}$) and experiences peak deceleration (MSL experienced 13 Earth g's). Due to friction with the atmosphere at hypersonic entry speeds, about 99% of the vehicle's kinetic energy is dissipated.

Once drag on the vehicle increases, the vehicle decelerates until the parachute deploy conditions are met. For MSL, deploy of the 21.35 m disk-gap-band parachute occurred at an altitude of about 10 km and a Mach number value of 1.75 [7]. During this event, the MSL entry vehicle decelerated from approximately 406 m/s to 78 m/s and removed the remaining 97% of its kinetic energy [6]. After parachute deploy, the vehicle ejects its heat shield and the radar begins to collect altitude data.

Following radar data collection, the vehicle is separated from the backshell and initiates powered descent and prepares for its final landing maneuver. MSL featured the novel "Sky Crane" maneuver. During Sky Crane, the descent stage lowered the rover to the surface of the planet using four steerable engines. Once

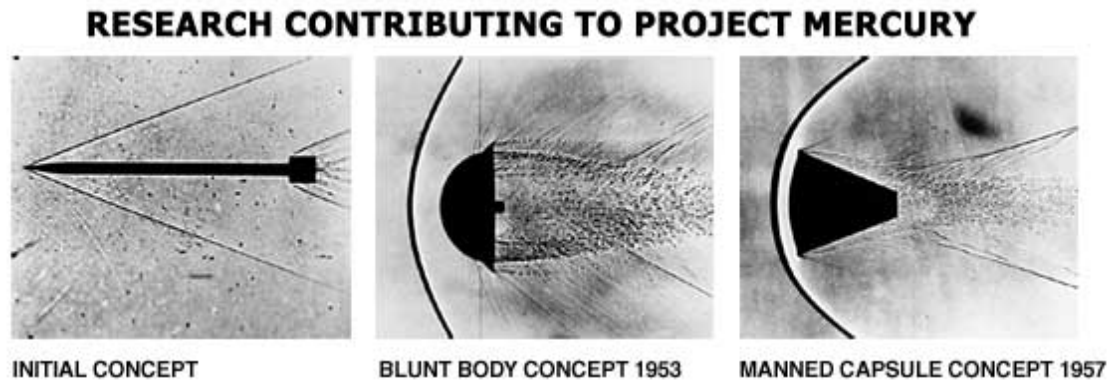


Figure 2.2: Shockwave separation for the models created for the Mercury re-entry vehicle concepts.

terminal velocity approached zero, the bridle joining the rover and the descent stage was cut and the rover was released to the ground.

2.1.1 Past Successful Mars Missions EDL Architecture

Atmospheric entry and re-entry systems were first explored during the Cold War. During this time, the Soviet Union and the United States were looking into expanding the capabilities of ballistic missiles and nuclear weapon technologies. Soon enough human exploration became a proxy for competition and efforts were aimed to develop entry vehicle systems that can decelerate the payload for safe landing and efficiently dissipate heat generated due to atmospheric drag. H.Julian Allen and A.J. Eggers, Jr. found that the most efficient heat shield that would require the least weight of coolant is obtained with a blunt-body shape (i.e. a shape with high pressure drag coefficient) [8]. As depicted in Figure 2.2, a blunt body separates the shockwave from the body, which helps minimize the maximum local heat transfer at the nose.

On August of 1976 the Viking mission became the first mission to land a spacecraft safely on the surface of Mars. Viking featured the 15 m diameter 70-degree sphere-cone forebody that provided a higher hypersonic drag coefficient than a slender body. The Viking EDL architecture also featured a 16m-diameter disk-gap-band parachute; and the SLA-561V thermal protection material. These technologies have served as the backbone components of all subsequent missions [1].

All successfully landed missions on the surface of Mars used the Viking era-derived technologies that were largely based on the lunar Surveyor and the Apollo mission. One reason is that end-to-end EDL system verification on Earth is not possible and thus, relies heavily on simulation for system verification and validation [1], [4], [6]. Consequently, every mission leading up to MSL used the 70-degree sphere-cone with the disk-gap-band parachute and the TPS of Viking with some modifications. These technologies increased in size as mass delivery requirements increased. Some aspects of EDL system qualification were thus achieved by similarity analysis to previous landed missions [1]. Figure 2.1 presents a summary of some of the technologies and key EDL metrics for successful landed U.S. missions on Mars.

2.1.2 Equations of Motion for Planetary Entry

Problems dealing with atmospheric entry are highly interdisciplinary due to the wide range of entry conditions experienced in this phase of a space mission. Vehicle systems entering a planet's atmosphere must be capable of dissipating kinetic and potential energy coupled with high entry speeds and altitude at

Table 2.1: Summary of EDL architecture of previous Mars missions [1].

Past Landed Missions (U.S.) on the Martian Surface								
Mission Name	Viking 1	Viking 2	MPF	MER-A	MER-B	Phoenix	MSL	
Year	1976	1976	1997	2004	2400	2008	200	
Entry velocity (km/s)	4.7	4.7	7.26	5.4	5.5	5.59	5.8	
Entry flight-path angle (deg)	-17	-17	-14.06	-11.49	-11.47	-13	-15.4	
Ballistic coefficient (kg/m^2)	64	64	63	94	94	65	115	
Entry guidance	None	None	None	None	None	None	Apollo guidance	
Heat Shield Geometry	70-deg cone	70-deg cone	70-deg cone	70-deg cone	70-deg cone	70-deg cone	70-deg cone	
Heat shield TPS	SLA-561	SLA-561	SLA-561	SLA-561	SLA-561	SLA-561	SLA-561	
Heat shield TPS thickness (in)	0.54	0.54	0.75	0.62	0.62	0.55	0.9	
Peak Heating (W/cm^2)	26	26	100	44	44	47	200	
Parachute diameter (m)	16	16	12.5	14	14	11.7	19.7	
Parachute deploy Mach no.	1.1	1.1	1.57	1.77	1.77	1.2	1.75	
Touchdown vertical velocity (m/s)	2.4	2.4	12.5	8	5.5	2.4	0.75	
Touchdown attenuation	3 crushable legs	4 crushable legs	4 π airbag	4 π airbag	4 π airbag	3 crushable legs	6 wheels	
Landed mass (kg)	590	590	360	539	539	382	899	

the entry interface. Moreover, vehicles must counteract torques and moments generated due to atmospheric forces by incorporating guidance commands to control lift and drag acting on the entry vehicle to ensure a safe landing [9].

Equations of Planar Motion

The simplest form of entry flight mechanics is given by the equations of planar flight over a spherical, non-rotating planet. The underlying equations for planar flight assume that the entry vehicle is subjected to drag only. In other words, the entry vehicle has no thrust and enters the atmosphere as a ballistic entry vehicle. Although entry vehicles can also generate lift forces during entry (e.g. Viking 1&2, MSL), planar trajectory analysis can lead to closed-form solutions that are convenient for assessing entry vehicle performance.

Figure 2.3 shows an entry body subjected to aerodynamic forces (lift and drag) and a geocentric gravitational field. In the representation provided, R_E is the planetary radius, θ is the central angle (e.g. or latitude if the trajectory is contained in a planet's meridian plane) and γ is the flight path angle (velocity direction with respect to the local horizontal).

As seen in Figure 2.3, the inertial plane $X^I Z^I$ contains the velocity vector throughout the motion. In the local plane $X^l Y^l$, Z^l is along the local vertical. In this representation the X^m -axis of the moving frame moves along the velocity vector. Equations 2.1 to 2.7 are the equations of motion for a ballistic entry vehicle.

$$D = C_D \rho V^2 S / 2 \quad (2.1)$$

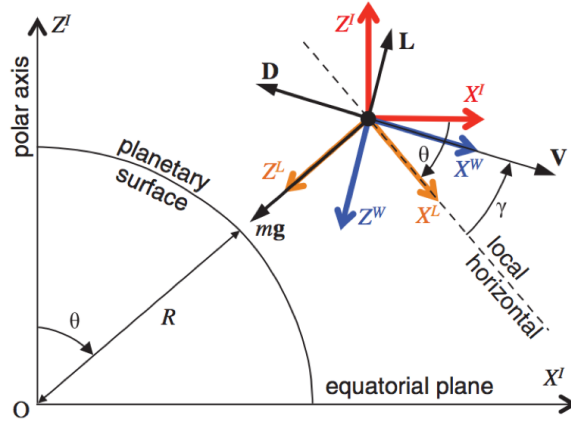


Figure 2.3: Planar flight FBD [12]

$$L = C_L \rho V^2 S / 2 \quad (2.2)$$

$$\frac{dV}{dt} = \frac{D}{m} + g \sin(\gamma) \quad (2.3)$$

$$V \frac{d\gamma}{dt} = -V \left(\frac{d\theta}{dt} \right) - \frac{\rho g C_L}{2\beta C_D} V^2 + g \cos(\gamma) \quad (2.4)$$

$$\frac{d\theta}{dt} = \left(\frac{V \cos(\gamma)}{R_E + h} \right) \quad (2.5)$$

$$\frac{dh}{dt} = V \sin(\gamma) \quad (2.6)$$

$$g = g_s \left(\frac{R_E}{R_E + h} \right)^2 \quad (2.7)$$

Equations 2.1 and 2.2 are the conventional drag and lift forces acting on the entry vehicle. The second two equations are Newton's Law of motion resolved in the in-track and cross-track directions. Equations 2.5 and 2.6 are kinematic relations resolved for the vertical and horizontal components of the velocity vector, respectively. Finally, equation 2.7 gives a relationship for gravitational acceleration. Knowing the definition of $d\theta/dt$, the lift and drag coefficients and the fact that the ballistic coefficient is defined by $\beta = m/C_D S$, equation 2.4 can be rewritten as shown in equation 2.8. Table 2.2 provides a legend for all variables involved in the equations of planar motion.

$$V \frac{d\gamma}{dt} = -\frac{L}{m} - \left(g - \frac{V^2}{r} \right) \cos(\gamma) \quad (2.8)$$

Variable Definitions for the Equations for Planar Flight	
Variable	Definition
D	Drag force
L	Lift force
C_D	Coefficient of drag
C_L	Coefficient of lift
V	Inertial velocity magnitude
S	Vehicle reference area
g	Gravitational acceleration
γ	Flight-path angle
θ	Central angle
ρ	Atmospheric density
β	Ballistic coefficient
R_E	Planetary radius
h	height above surface
$r = R_E + h$	Radius from planetary center

Table 2.2: Variable definition for the equations of planar motion during ballistic flight.

By accounting for atmospheric density variations (Equation 2.9) and by expressing the drag coefficient in terms of the ballistic coefficient, equations 2.3 to 2.6 can be expressed as the following closed-form solutions illustrated in equations 2.10 to 2.12. These equations have no analytical solution and can only be resolved by means of numerical integration [12]. It is worth recalling that these equations account only for aerodynamic and gravitational forces (e.g. no thrust or lateral forces); drag and lift are assumed to be constant; and the atmospheric density varies exponentially with constant temperature.

$$\rho = \rho_0 e^{-\beta h} \quad (2.9)$$

$$\frac{dV}{dt} = \rho e^{-\beta h} V^2 \left[\frac{S C_D}{m} \right] - g_s \left[\frac{R_E}{R_E + h} \right]^2 \quad (2.10)$$

$$V \frac{d\gamma}{dt} = \rho e^{-\beta h} V^2 \left[\frac{S C_D}{m} \right] \left[\frac{L}{D} \right] - \left[g_s \left[\frac{R_E}{R_E + h} \right]^2 - \frac{v^2}{R_E + h} \right] \quad (2.11)$$

$$\frac{d\theta}{dt} = \left(\frac{V \cos(\gamma)}{R_E + h} \right) \quad (2.12)$$

$$\frac{dh}{dt} = V \sin(\gamma) \quad (2.13)$$

Allen-Eggers' Approximations for Ballistic Entry

During the mid-1900's, analytic approximations for lifting and ballistic entry were developed by Sängers and H. Julian & A.J. Eggers, Jr. These approximations would aid mission designers evaluate entry trajectories and vehicle performance with minimal computational requirements. In particular, the Allen-Eggers solution for re-entry vehicles was considered a breakthrough in the field of engineering. The analytical solutions obtained by Allen and Eggers suggest that that aerodynamic heating on an entry vehicle during peak deceleration is minimized by employing shapes with high pressure drag [8]. Furthermore, their analytical solution demonstrated that at steep flight-path angles gravitational forces can be neglected. These findings yielded closed-form analytical expressions for peak deceleration, peak heating as well as altitude and velocity at maximum deceleration [12].

Allen and Eggers made some additional assumptions and simplifications to the equations of planar motion as follows:

1. Atmospheric entry is ballistic (i.e. no lift is generated)
2. Gravitational and centrifugal terms can be neglected
3. Constant flight-path angle (i.e. trajectory is a straight line)

4. Initial density ratio (or specific gravity) can be neglected at entry interface

Equations 2.13 to 2.15 are the analytic Allen-Eggers' approximations for ballistic entry. Equation 2.14 is the simplified differential equation that neglects the effect of gravity given that the magnitude of the drag force is substantial when compared against the effects of gravity [8][12]. Equation 2.15 is the analytic solution using the altitude reference at the entry interface. Equation 2.16 is the approximate solution for the velocity profile.

$$\frac{1}{V} \frac{dV}{dt} = \frac{\rho_0}{2\beta \sin(\gamma_{EI})} \left[\frac{SC_D}{m} \right] e^{-\beta h} \frac{dh}{dt} \quad (2.14)$$

$$\frac{V}{V_{EI}} = e \left(\frac{\rho_0}{2\beta \sin(\gamma_{EI})} \left[\frac{SC_D}{m} \right] (e^{-\beta h} - e^{-\beta h_{EI}}) \right) \quad (2.15)$$

$$\frac{V}{V_{EI}} = e \left(\frac{\rho_0}{2\beta \sin(\gamma_{EI})} \left[\frac{SC_D}{m} \right] e^{-\beta h} \right) \quad (2.16)$$

Knowing the density relation and the analytic approximate solution, the altitude at which maximum deceleration occurs for a ballistic entry vehicle is presented in equation 2.17. The velocity at maximum deceleration, as shown in equation 2.18 is a constant that results to be a fraction of the entry velocity and is roughly $0.6V_{EI}$. Finally, equation 2.19 shows that the magnitude of the peak deceleration is determined by the entry velocity and flight-path angle [8].

$$h_{a_{max}} = \frac{1}{\beta} \ln \frac{C_D \rho_0 S}{\beta \sin(\gamma)} \quad (2.17)$$

$$V_{a_{max}} = \frac{V_{EI}}{\sqrt{e}} \quad (2.18)$$

$$a_{max} = \left(\frac{\beta V_{EI}^2}{2e} \right) \sin(\gamma) \quad (2.19)$$

2.1.3 Aeroheating During Planetary Entry

Heat flux at the stagnation point is modeled using the Sutton-Grave correlation, shown in Equation 2.20. This equation gives the convective heat transfer q_{conv} per unit area. Parameters such as the constant k and density ρ in the Sutton-Graves correlation are based on the planetary atmosphere for a velocity of V_∞ . By inspecting Equation 2.20, it can be observed that peak heat flux is minimized by vehicles which large nose radius [14].

$$q_{conv} = k \sqrt{\frac{\rho}{R_n}} \left(\frac{V_\infty}{1000} \right)^3 \quad (2.20)$$

Similar to the solutions at peak deceleration developed by Allen-Eggers, the solutions for peak heat rate are derived and result in altitude at peak heat rate and peak heat rate described by Equations 2.21 - 2.22 [12].

$$h_{max} = k * \ln \left(\left(\frac{-3}{\beta} \right) \left(\frac{SC_D}{m} \right) \frac{\rho}{\sin \gamma_{EI}} \right) \quad (2.21)$$

$$q_{max} = k \sqrt{\left(\frac{\beta}{3e} \right) \left(\frac{m}{SC_D} \right) \frac{\sin \gamma_{EI}}{R_n}} V_{EI}^3 \quad (2.22)$$

2.2 Limitations of EDL Technologies

As the interest in high-scientific return missions to Mars, mass delivery requirements increase. One immediate consequence is the increase of the ballistic coefficient of the entry vehicle. Consequently, the magnitude of peak deceleration

increases, the altitude at which peak deceleration occurs decreases, and peak heat rate increases. Whereas robotic missions can potentially endure high g-loads and high heat loads, landing humans on Mars will require peak decelerations of 5 Gs or below, and decreased heat loads [1].

One alternative for solving the issue of altitude at which peak deceleration occurs is to reduce the ballistic coefficient of the entry vehicle. This approach is applicable to lifting and non-lifting atmospheric entry schemes. However, there are only three variables in play that can be adjusted. The first alternative would be to decrease the mass of the payload delivered. The first alternative is unfeasible given that the objective is to deliver heavier payloads safely on Mars. For example, landing humans on Mars could potentially require delivering payloads up to 100 tons. For robotic missions mass is highly constrained by the mission science objectives and thus the instrumentation, making the only alternative for reducing mass to sacrifice scientific instrumentation. Mass could be further reduced by using a lighter TPS material that has equivalent performance to the SLA-561 TPS or better. However, such approach would require space qualification of a new heat shield.

Another alternative would be to increase the drag coefficient of the entry vehicle. However, the task of finding the optimal geometry that achieves the highest hypersonic drag coefficient has already been solved. As stated by Braun et. al., a different aeroshell configuration to the 70-degree sphere-cone geometry would not have a significant impact on hypersonic drag coefficient and thus, cannot be relied on to improve EDL performance.

Finally, the remaining alternative is increasing the diameter of the aeroshell. However, the maximum allowable diameter is highly constrained by the launch

system employed. Currently, the maximum allowable diameter is 4.6m by using an Atlas 5-class launch system. MSL has already achieved this limit.

Due to the constraints imposed on EDL architectures, landing larger payloads than MSL will require the addition of low-mass hypersonic decelerators. At the moment NASA has proposed using hypersonic inflatable aerodynamic decelerators (HIAD), supersonic inflatable aerodynamic decelerators (SIAD) and lifting variants of these technologies [15][16].

CHAPTER 3

SIMULATIONS FOR ENTRY, DESCENT AND LANDING

Because replicating the Martian environment on Earth is infeasible, it is necessary to analyze a wide range of EDL architectures to assess performance with high-fidelity simulations of performance, cost, schedule, and risk under uncertainty. For previous Mars missions, such as MSL, engineers used the Program to Optimize Simulated Trajectories II (POST2) to simulate a variety of entry conditions under many model parameterizations (e.g. gravity, planetary geometry, atmospheric, aerodynamic, control system, guidance, and navigation models) and evaluate performance, mission-level feasibility, and design trades, among other capabilities [4][6]. An overview of the POST2 simulation model as a function of entry conditions (e.g. entry flight path angle α , ballistic coefficient β) is shown in Figure 3.1. Entry vehicle dynamics are modeled as a function of entry conditions subjected to the gravitational environment at the entry location and vehicle aerodynamics- which are dependent on the atmospheric environment. Following the flow presented in Figure 3.1, entry conditions and vehicle dynamics are used in parallel to model the navigation system to estimate the vehicle's attitude and position, which is then provided to the guidance algorithm. Next, guidance provides to the control system the required commands for re-orientation maneuvers of the vehicle. As a result, new vehicle dynamics are estimated as a function of the control system and approach navigation.

Due to the limitations for Earth-based testing, some models in POST-II make use of wind tunnel testing (e.g. parachute deploy conditions) and computational fluid dynamics models (e.g. aerodynamics, aerothermodynamics) whereas some models rely on data collected from previous orbital missions on

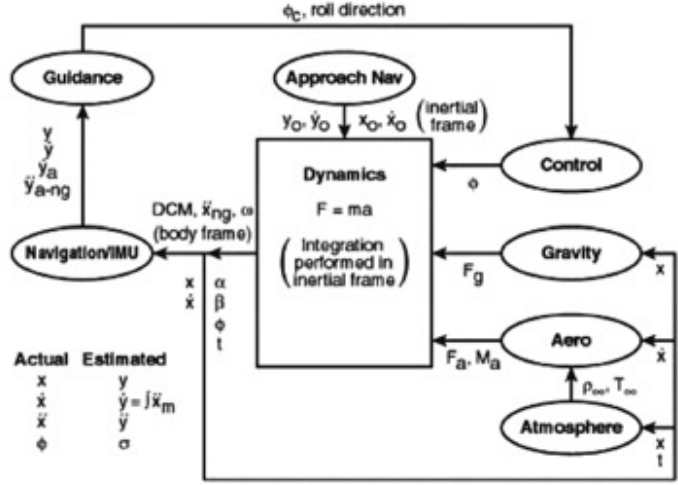


Figure 3.1: POSTII simulation models and data flow [4].

Mars (e.g. atmospheric models, elevation maps). Figure 3.2 presents the models that have been incorporated into POST-II up to the present day. Typically, existing models in POST-II are modified and used to evaluate top-level trades of new designs during concept studies and technology development. As new system requirements are being developed, complexity of the models increases up to day-of-entry operations.

3.1 POST-II-based Monte Carlo Dispersion Analysis

Monte Carlo simulation techniques are commonly employed to solve deterministic problems whose inputs are randomly sampled from probability distributions that describe the behavior of certain parameters [17] [18]. For EDL, simulation models implemented are purely deterministic. For a fixed set of input conditions, the resulting outputs and behavior are always be the same. Hence, due to the uncertainty in what the entry conditions will be at the day-of-entry,

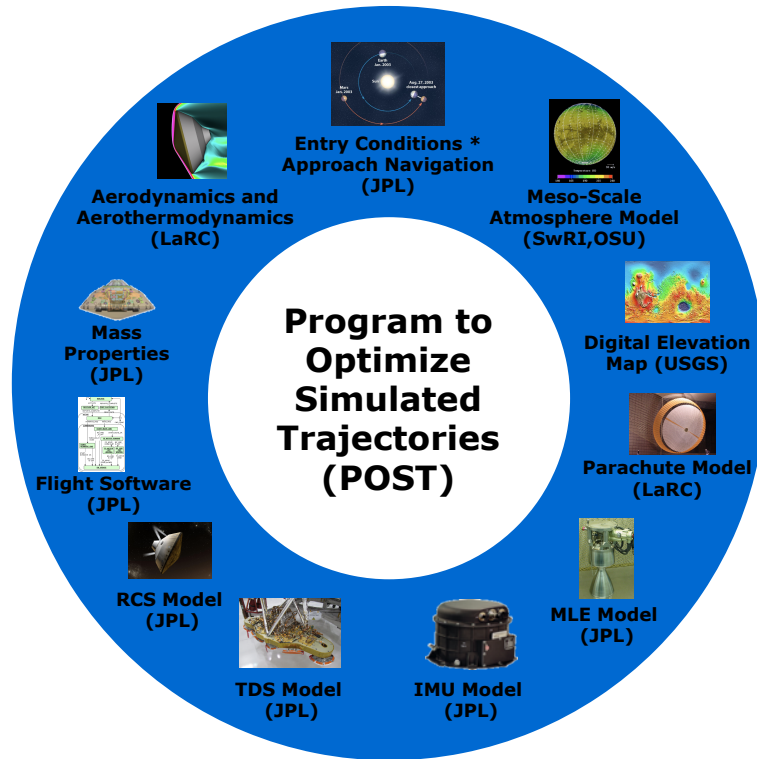


Figure 3.2: Current models evaluated in POSTII

POST-II simulations are executed by sampling a wide range of entry conditions to capture off-nominal cases that might present a risk to mission success. Results obtained from EDL POST-II-based Monte Carlo techniques are critical for identifying areas of risk associated with certain designs or mission phases, and for quantifying the robustness of a given EDL architecture by randomly varying entry vehicle configurations (e.g. lift-to-drag ratio, entry flight path angle) and environmental conditions (e.g. entry interface, atmospheric conditions) up to the day-of-entry. Any metric that does not comply with mission success criteria has to be examined and re-evaluated by the model developers and mission designers [4].

For MSL, hundreds of input parameters are sampled and a total of 8,001 test cases are executed in a single Monte Carlo simulation. For each simulation case

selected parameters are always randomly sampled. However, in the attempt of identifying stressing cases, some inputs are occasionally left at their nominal values. These parameters are varied from the nominal values provided and the respective $3\text{-}\sigma$ dispersion values. As models are updated and input variations are provided, test cases are executed and examined between the model developers, mission design team, and the team in charge of examining POST-II simulation results.

Figure 3.3 shows the the typical EDL validation and verification process between model developers and simulators. As it can be seen, both model and simulation developers work collaboratively to ensure that the code implemented reproduces the expected results. Once simulation case is implemented and executed in POST-II, the design team has to analyze the dataset produced. Often the first task is to generate a scorecard. The scorecard can be used to make a quick assessment on the performance of the architecture evaluated in a specific simulation. Furthermore, the design team can rapidly identify what metrics present unusual or undesired behavior.

Due to the nature of the simulation executed, the EDL team is also interested in analyzing the statistics of key metrics (e.g. timeline margin, fuel consumption) to identify whether the upper or lower percentiles satisfy system requirements. For example, if the maximum fuel consumption permitted is 300 kg, the team is interested in seeing that the 99-percentile is below the requirement threshold. To do so the team has to load datasets and execute functions to obtain the statistical information of interest. Alongside visual representations are generated for examining closely results and outliers. The EDL team makes use of hundreds of plots that are generated to examine margins, ellipses, and statis-

tical information. All of these data products generated are used by the design team to assess risk and evaluate system performance.

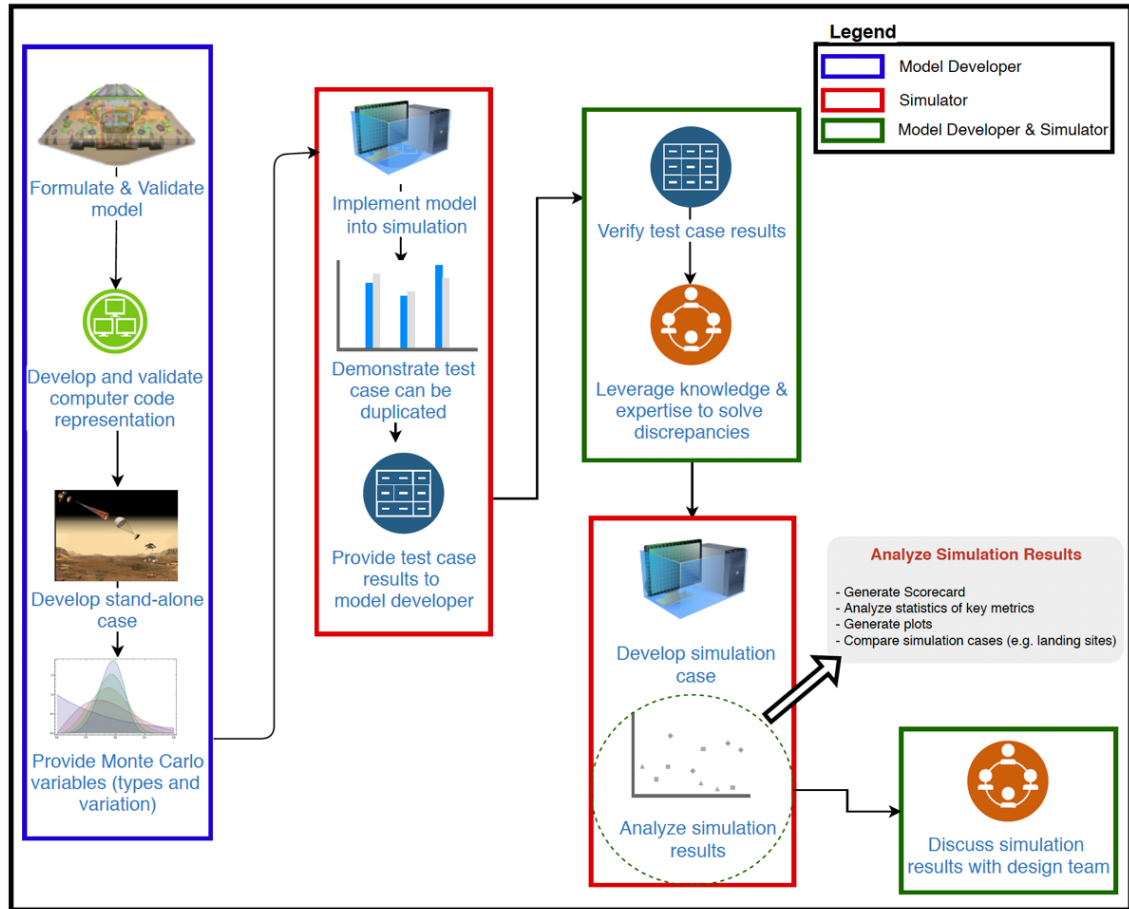


Figure 3.3: Current models evaluated in POSTII

Inevitably, as model complexity increases throughout a mission's life-cycle, the size of the data sets that have to be examined increases. Each test case in a simulation can generate from hundreds to thousands of output variables that capture critical events in the EDL sequence as well as event records that are evaluated against the in-flight data generated to assess model accuracy.

3.2 Sensitivity Analysis of POST-II Based Monte Carlo Dispersion Analysis

EDL trajectory is sensible to four all-encompassing EDL parameters: entry flight path angle, ballistic coefficient, entry velocity, and lift-to-drag ratio. In reality, however, the EDL architecture is fraught with uncertainties and additional sensitivities due to the inherent complexity of each vehicle configuration. Consequently, assessing sensitivity of figures of merit remains a challenge in EDL architecture analysis.

The most common approach for EDL teams to assess sensitivities in EDL architecture analysis is the methodology of the *one-at-a-time* approach. In other words, sensitivities in a complex architecture are identified by varying input variables in one model, while maintaining all other metrics at their nominal values. For example, to assess the sensitivity of aerodynamics in the POST2 trajectory analysis, a Monte Carlo simulation is executed with uncertainty in aerodynamics metrics while maintaining all other models (e.g. gravitation, guided entry) at their nominal values. This process is repeated for all models and sensitivities to each model are compared by means of scatter plot and 3σ dispersion analysis. Whereas this approach provides a straight-forward manner to identifying model sensitivities, this method fails to capture the effect of interactions in all input variables and uncertain parameters [19].

CHAPTER 4

INTELLIGENT DATA UNDERSTANDING

4.1 Intelligent Data Understanding (IDU) Technologies

As mentioned in Chapter 3, throughout a mission’s lifecycle the number of simulations can reach the order of thousands, given that different entry conditions and model parametrizations are evaluated to assess performance, mission-level feasibility, risks, and design trades. Whereas the subject matter expert can leverage his or her expertise to analyze these datasets to identify features of interest, the next generation of EDL systems might present unprecedented challenges. Consequently, EDL architecture analysis can benefit from computational advances to reduce analysis cycle time, minimize architecture lifecycle costs and achieve mission success.

To address challenges in architecture analysis, NASA suggests the advancement of the state of the art in analysis capabilities to enable informed architecture trades and technology development decisions. In this work, we suggest incorporating IDUs as technology, a form of information processing, that can benefit architectural analysis for EDL to reduce analysis lifecycle time, minimize lifecycle costs, and reduce risk. IDU technology can provide capabilities to “*to automatically mine and analyze datasets that are large, noisy, and of varying modalities, including discrete, continuous, text, and graphics, and extract or discover information that can be used for further analysis or decision making*”. Incorporating IDU technologies for EDL architecture analysis can aid the design team to rapidly evaluate EDL performance metrics, identify high information content data (e.g. anomalies, novelties, events) and improve inferences made using single data

sources [22].

4.1.1 State of the Art of IDU Technologies

Up to the present day, IDU technologies for space systems applications have been employed for on-board data processing and analysis capabilities. For example, NASA's Space Cube 2.0 incorporated on-board processing capabilities to provide the system with first-responder real-time awareness, enable multi-platform collaboration, and to conduct ground-based data processing on-board [20]. Similarly, NASA's Earth Observing 1 (EO-1) has featured IDU technologies for on-board planning and scheduling tasks. Onboard processing enables the system to detect science events and respond accordingly. For example, EO-1 can perform on-board cloud detection and data targeting [21].

Although on-board smart sensing techniques have been proven useful for event detection on-board spacecraft, the percentage of decisions that can be made autonomously for current technologies is between 20-30% of all decision instances. The technology goal for IDUs for on-board data processing is to increase autonomy of these systems [22].

Other advances for automation and automated data mining of data sets include the Automatic Statistician. This tool explores multiple statistical models to discover explanations for a given dataset and returns a natural language-form report explaining the data. However, like other similar technologies, this method fails to capture and incorporate human knowledge into the loop. One reason this methodology and approach might result in a drawback of the tool is that because all aspects of the data mining process are automated, the end

user does not have the opportunity to explore specific regions of interest in a given data set. Hence, some of the information provided is not of interest to the end user, resulting in additional work to find the information of interest. Consequently, some users might benefit from the ability to select a family of models for exploring a portion of the dataset being examined. The interaction between the human and the tool can thus be enhanced by means of dialogue and interactive tools. Consequently, there is a potential benefit in advancing the state of the art for data mining tools with the human in-the-loop to improve the process [23].

Up to the present day, IDU technologies have been employed for operational processes to define the knowledge involved in the given process but its use in performance analysis has not been explored. Furthermore, as mentioned by Gharhamani et. al. in Reference [23], tools for feature extraction and knowledge discovery such as the "Automated Statistician" can benefit from user interaction capabilities. However, systems for such capability must be supported by the underlying system, making cognitive assistants an alternative to provide such requirement due to the underlying interactive process between humans and machines. Therefore, in this thesis we explore the roles of utilizing a cognitive assistant as the platform for the next generation of IDU technologies for performance analysis of EDL systems. Incorporating IDU technologies for data analysis can help experts identify problems with a design earlier in a mission lifecycle, which may lead to less rework, large cost savings and increased probability of success.

4.2 Cognitive Assistants for Space Applications

Cognitive assistants have been explored as a platform to incorporate intelligent decision and analysis technologies to provide decision-making support under uncertainty [24]. Decision decision-making processes can be of individual or collaborative nature. However, one commonality between both scenarios is that making informed and well thought decisions require the task of gathering information, weighing the evidence and making a choice among the alternatives [26]. Sources of information can come in the form of expert knowledge or can come as a result of rigorous analysis. Often this knowledge comes from principles or heuristics experts in the field use for ensuring mission success. What makes these heuristics critical is that they are not implicit during the analysis and rather they are knowledge gained from experience. Consequently, introducing a cognitive assistant into the work domain can provide expert knowledge alongside computational analysis capabilities through a collaborative process.

There are two aspects of a cognitive assistant in which we are interested. The first is using a cognitive assistant as a form of *decision support system* that can take advantage of computational tools, such as IDUs. What makes a CA distinct from traditional decision support systems is that they attempt to emulate human behavior as close as possible. Generally CA systems possess inference capabilities that use of machine learning and data mining tools, to achieve better performance under a variety of scenarios. Combining both technologies can add more rigor and exhaustiveness to the analysis process. Second, a cognitive assistant can serve as an additional *team member* that helps automated some steps or specific tasks. Furthermore, the assistant can also provide comments

or recommendations in a manner similar to team-like collaboration. Therefore, introducing a cognitive assistant into the work domain can enhance the process, by providing feedback obtained from advanced data analysis capabilities that aid the team in decision-making tasks under uncertainty.

A CA can help examine the consequences of decisions leading to the discovery of critical information on a design. More specifically, cognitive assistants could help respond questions on the stakes and probabilities of success of a particular design [24]. For example, the CA can be used to evaluate the consequences of a particular technology selection in EDL performance (e.g. propellant consumption, landing accuracy) and probability of mission success. In particular, a CA can be useful for identifying critical metrics in a particular architectural design that have pivotal consequences on the system under study but could be easily missed by experts. Hence, incorporating a CA into the loop that possesses an expert knowledge-base, analysis and inference capabilities combining the problem solving expertise from subject-matter experts can lead to conduct better informed architectural design decisions in an early stage of a design. Furthermore, a CA can provide analysis capabilities that reduce the effort of searching through hundreds of data sets containing thousands of cases each to identify features of interest that drive further analysis and decision making.

During recent years, efforts have been made to create methodologies for the development of knowledge-based cognitive assistants. One example is COGAS, a cognitive assistant that supports NAVY operators during air target identification. This CA obtains information from on-board instrumentation to visualize and track targets and describes opportunities for action in the case of hostile threats [25]. As for missions beyond Low Earth Orbit (LEO) in particular, one

of the most recent projects is the development of requirements for a CA that can provide on-board support to space crew during off-nominal conditions in scenarios where communication between mission control and the spacecraft is delayed or limited [27]. Although action protocols tend to be defined in the case of off-nominal situations, astronauts may still face off-nominal situations that are not well established. Therefore, astronauts may face decision-making tasks under uncertainty and need to make informed decisions to ensure the safety of the crew and the vehicle to ensure mission success. One well-known example is the Apollo 13 emergency return to Earth due to the loss one of the three fuel cells and an oxygen tank. By following emergency protocols and with the assistance of the mission control center, all three astronauts were able to perform an emergency maneuver and landed safely on Earth. During delayed or limited communications with ground station, a cognitive assistant can provide assistance while communications with Earth become available. Utilizing a CA for spacecrew support requires that the assistive system possesses functional capabilities for data collection, analysis, communication of critical issues and guidelines and how to approach the problem. Furthermore, Neerinx et al. has established a cognitive engineering (CE) methodology for integrating task procedures, tools, and methods into user interface design to test human-machine collaboration in space.

The remainder of this chapter introduces Daphne, a cognitive assistant that can be used in-the-loop for architecture analysis. Daphne is a cognitive assistant that specializes in architecting Earth observing (EO) satellite systems. Currently, Daphne can analyze satellite-specific problems and can provide critiques that address strengths and weaknesses for a particular design [28]. This research seeks to explore intelligent data understanding technologies by extend-

ing Daphne to be used as a platform for EDL architecture analysis by providing information from a particular architecture to the end user using multiple sources of knowledge. With Daphne, we want to explore the ability of an intelligent assistant to access and analyze EDL datasets and identify and communicate critical issues to the human.

4.3 Daphne Architecture

Figure 4.1 shows Daphne’s architecture for EO satellites and its four primary capabilities: 1) the Analyst, 2) the Critic, 3) the Historian, and 4) the Explorer. The architecture of Daphne consists of four layers. The front ends serve for the user to interact with Daphne and direct requests to the respective back-ends [31] that extract the features of interest from the data sources. The analyst is in charge of answering questions through queries about a particular design. For example, the user can ask questions about past designs such as: “Why does this design have this value for X performance metric?” The Historian provides historical information on previous missions and can be used during the design process to check whether selected parameters and the design being evaluated are similar to those of past designs. Example questions include “What is the most common orbit for ice cloud detection?”. The critic skill takes the proposed design and provides feedback to the users about the strengths and weaknesses of the design. One feature of the critic skill is that it also provides suggestions on how to improve the design. Finally, the Explorer executes a genetic algorithm in the background. As Daphne finds solutions that improve the Pareto front, Daphne asks the user if she/he wants these solutions in the current dataset. The latest addition to Daphne is the EDL analysis capability.

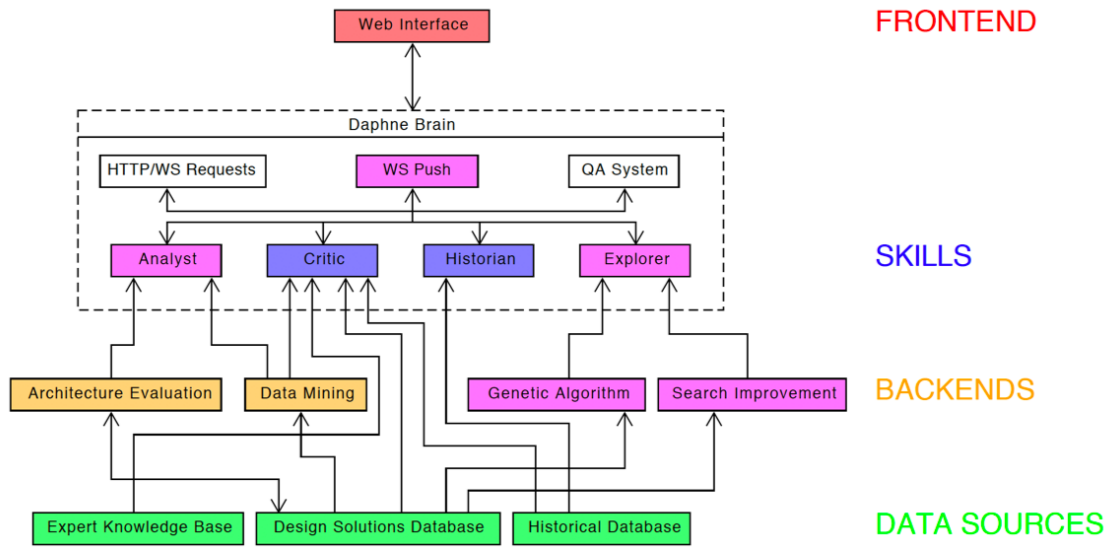


Figure 4.1: Daphne Brain Architecture

4.4 Daphne for EDL

Daphne has been selected as the platform for an EDL intelligent assistant for architecture analysis. As shown in Figure 4.1, Daphne is structured as a 4-layered system. First, Daphne has a front-end that allows the user to interact with the system (e.g. queries, commands). Second, in the Daphne brain is the bridge between the user front-end and the back-ends. In the Daphne brain, requests are sent to each of the skills. Responses are pushed back to the users in the front-end. In the third layer, the back-ends contain the capabilities for resolving the queries. Back-ends make use of the data sources available in the fourth layer to obtain the data requested.

During the rest of this chapter we describe the Daphne architecture, focusing on those capabilities adapted for the EDL intelligent assistant.

4.4.1 Frontend

Daphne makes use of a visual interface that allows the user to make queries or commands to Daphne. At the moment, Daphne for EO contains a section where the user can write her/his query to Daphne. The response is provided to the user in natural language-form in another panel. Below the question panel the web interface contains a tradespace exploration panel that allows the user to visualize the objective space of the alternative architectures. Below the tradespace exploration panel is the Design Inspector, which displays the decisions for the architecture being visualized.

At the moment the Daphne EDL assistant does not have its own web-based interface. However, the Daphne EO interface has been used to test the data extraction queries for EDL as well as the EDL commands.

4.4.2 Daphne Brain

As for the Critic and Analyst in Daphne for EO, the EDL assistant accepts question in natural language form. As described in Figure 4.2, requests from the user are classified into either commands or questions. In this case we assume that the user asks Daphne "What was the entry velocity for MSL?" Once the request is classified, Daphne then proceeds to classify the request by type (e.g. EDL). This task is achieved by means of a Convolutional Neural Network (CNN). The existing algorithm in Daphne was adapted to classify questions regarding EDL. Reference [28] contains a more detailed description of the CNN model implemented. It must be noted that at the moment all of the EDL questions are contained within the EDL "skill", however this capability is not meant to be a skill

like the analyst or the critic considering that the current queries for EDL are of analyst and historian nature. Furthermore, the intention is to make use of these skills for EDL simulation datasets. However, for testing the first prototype, all queries are stored under the EDL "skill".

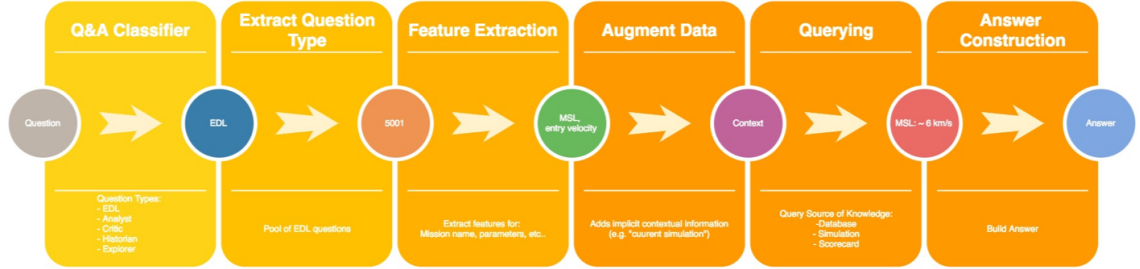


Figure 4.2: Question classification in Daphne.

Following question classification by type, Daphne searches for the information requested in the query. JSON file templates available in Daphne, such as the one in Figure 4.4, specify the name/value pairs required to respond to a particular query and are used to search for the features requested. For the query made, we want to extract two features: mission and parameter. Feature extractors match the sentences to lists of known values for the requested information. Daphne's implementation of the statistical model provided by Sellers et al., algorithm accounts for mistakes (e.g. typos) in the user's request [30]. In this case, features are extracted from the historical database in the query section of the template. Finally, after features are extracted, results are embedded into the template response. The response is then returned to the user at the front end with in the voice or visual response template.

EDL Queries

Considering the breadth and depth of the subject of EDL, experts in the field provided a list of frequent questions that arise when analyzing a Monte Carlo output dataset as well as common tasks. From the list of questions created, ten question types were identified. The preliminary question types are listed in Figure 4.3. Given that the capabilities implemented in this thesis focused on data extraction and analysis capabilities for a cognitive assistant, key functional requirements for a CA [27], emphasis was given to queries that request values or parameters from the historical database, simulation files, and score-cards. Achieving the task of obtaining metrics of interest provides the first step for expanding the capabilities of Daphne for EDL architecture analysis such as combining knowledge from sources of data to conduct high fidelity analysis that enable informed architectural decisions. The queries that the Daphne EDL skill can address at the moment are:

- Find value of a parameter for a mission or simulation
- Compare values of a parameter across missions or simulations
- Plot evolution of a parameter over time, possibly across missions
- Plot histogram
- Plot quad-plot
- Plot statistics of a parameter (e.g., cdf, quantiles)
- Provide specific statistics (e.g., max, min, mean, 99% percentile) of a parameter
- Is X correlated with Y?
- How is this result different from mission X?
- Identify a stressing case

Figure 4.3: Envisioned question types for Daphne.

1. What is parameter << *parameter* >> for mission << *mission* >>?
2. What are the simulation files available for mission << *mission* >> ?

3. Load the simulation file << *file* >>.
4. For mission << *mission* >>, calculate the statistics for << *parameter* >>.
5. What is the value of parameter << *parameter* >> from this simulation?
6. For this simulation file, generate the scorecard.
7. From the scorecard, what are the POST2 results for << *metric* >> ?
8. What metrics in the scorecard are flagged ?
9. What metrics in the scorecard are out of spec?

```
{
  "type": "db_query",
  "params": [
    { "name": "mission", "type": "edl_mission", "options": "", "mandatory": true },
    { "name": "parameter", "type": "parameter", "options": "", "mandatory": true }
  ],
  "query": {
    "always": "session.query(models.Mission, models.Entry).join(models.Entry).filter(models.Mission.name.ilike('%${mission}%'))",
    "opt": [],
    "end": "",
    "result_type": "single",
    "result_fields": {
      "parameter_result": "row.${parameter}"
    }
  },
  "voice_response": {
    "type": "single",
    "template": "For mission ${mission} the ${parameter} is ${parameter_result}."
  },
  "visual_response": {
    "type": "text",
    "from": "single",
    "template": "For mission ${mission} the ${parameter} is ${parameter_result}."
  }
}
```

Figure 4.4: Query template for historical database.

Skills

The Daphne brain for EO contains four essential skills: 1) Analyst, 2) Critic, 3) Historian, and 4) Explorer. These skills add functionalities to Daphne. When the user makes a query, the QA system directs the question to the appropriate back-ends or data sources. Considering that the efforts in this work

were establishing the baseline for an EDL intelligent assistant, emphasis was given to the Historian and Analyst skills given that we want to be able to extract simulation data sets and analyze it.

Daphne EDL as Analyst

The objective of the Analyst is to provide responses on a particular EDL architecture problem. At the moment, the Daphne EDL analyst is not implicit as for EO. However, when Daphne obtains an EDL query, she can handle questions regarding historical information, whether a particular simulation satisfies performance requirements and can provide basic statistics on EDL parameters and metrics. A brief description of these roles are given below.

- **Historical Information** For EDL architecture analysis tasks, the user can ask Daphne questions about past EDL architectures. For EO architecting problems historical information is more descriptive. For example, historical information on EO missions can regard of instrument-orbit placement or instruments used for a particular measurement. For EDL, engineers are more interested in obtaining information on the value of critical EDL metrics. For example, "What was the peak deceleration for MSL?", "What was the entry velocity for MPF?" Other historical information of interest are the technologies and mechanisms employed in previous EDL missions. The Analyst can extract from the historical database: "What TPS did MSL use?" or "What was the type of entry for MSL?". As follow-up questions the user can ask "What was the heat shield diameter for MSL?", "What was the TPS material used?" Rather than having to make database queries, Daphne can do this task for the user.

- **Performance Requirement Satisfaction** When looking at simulation results, EDL teams are interested in looking at what metrics are within the acceptable threshold requirements. The analyst can be used to respond this question on a particular data set. If a scorecard for a particular EDL architecture is available, the analyst can extract what metrics are satisfying the requirements. Otherwise, the user can ask Daphne to automatically generate a scorecard on a particular dataset. The user can ask Daphne "From the scorecard, what metrics are flagged?", "From the scorecard, what metrics are out of spec?" Daphne returns a list of metrics that are flagged or out of spec to the user along with the simulation value compared against the requirement value.
- **Statistics of Metrics** Once the EDL POST II-based Monte Carlo simulation is executed it is common practice to take a glimpse at statistics of key performance metrics. In the current implementation of the Daphne EDL assistant the user can ask about the statistics of the dataset being examined using natural language form. For example, common questions for Daphne would be: "What are the statistics for peak deceleration?", "What are the statistics for altitude of parachute deploy?" In Daphne, this part of the process is automated and requires no manual effort from the experts to navigate, locate, load, and execute functions to evaluate statistics of a simulation.

4.4.3 Backends

At the moment, Daphne for EDL does not make use of any machine learning algorithms for evaluating an EDL architecture. For EO, Daphne makes use of an

architecture evaluation algorithm, data mining capabilities, a genetic algorithm and a clustering algorithm. For EDL, as a part of future work we plan on incorporating capabilities for architecture evaluation and data mining for obtaining insight on features of interest in an EDL simulation dataset.

4.4.4 Data Sources

EDL Historian

One of the sources of knowledge for analysis of EDL architectures is the historical database. The historian implemented for this role, or “skill” contains a database of previous successfully landed missions on Mars. The database contains nominal mission parameters for previous missions and contains a set of simulation files corresponding to MSL and the ongoing Mars 2020 mission. The EDL database was implemented as an object-relational database management system (ORDMS). Such database provides a bridge between relational and object-oriented paradigms. The standard selected for managing information in the database is the Structured Query Language (SQL) through the PostgreSQL software.

The resulting database was built in Python using the SQLAlchemy toolkit and served as an interface between the database and PostgreSQL. In the current model, one-to-many relationships were incorporated to connect fields in a given class (i.e., table) to another table. In other words, the current model uses hierarchical relationships. However, considering that not all relationships are of this nature (e.g., different missions or segments can share the same attitude control mechanism), many-to-many relationships can be discovered. Thus, many-to-

many relationships can be incorporated to account for additional complexity.

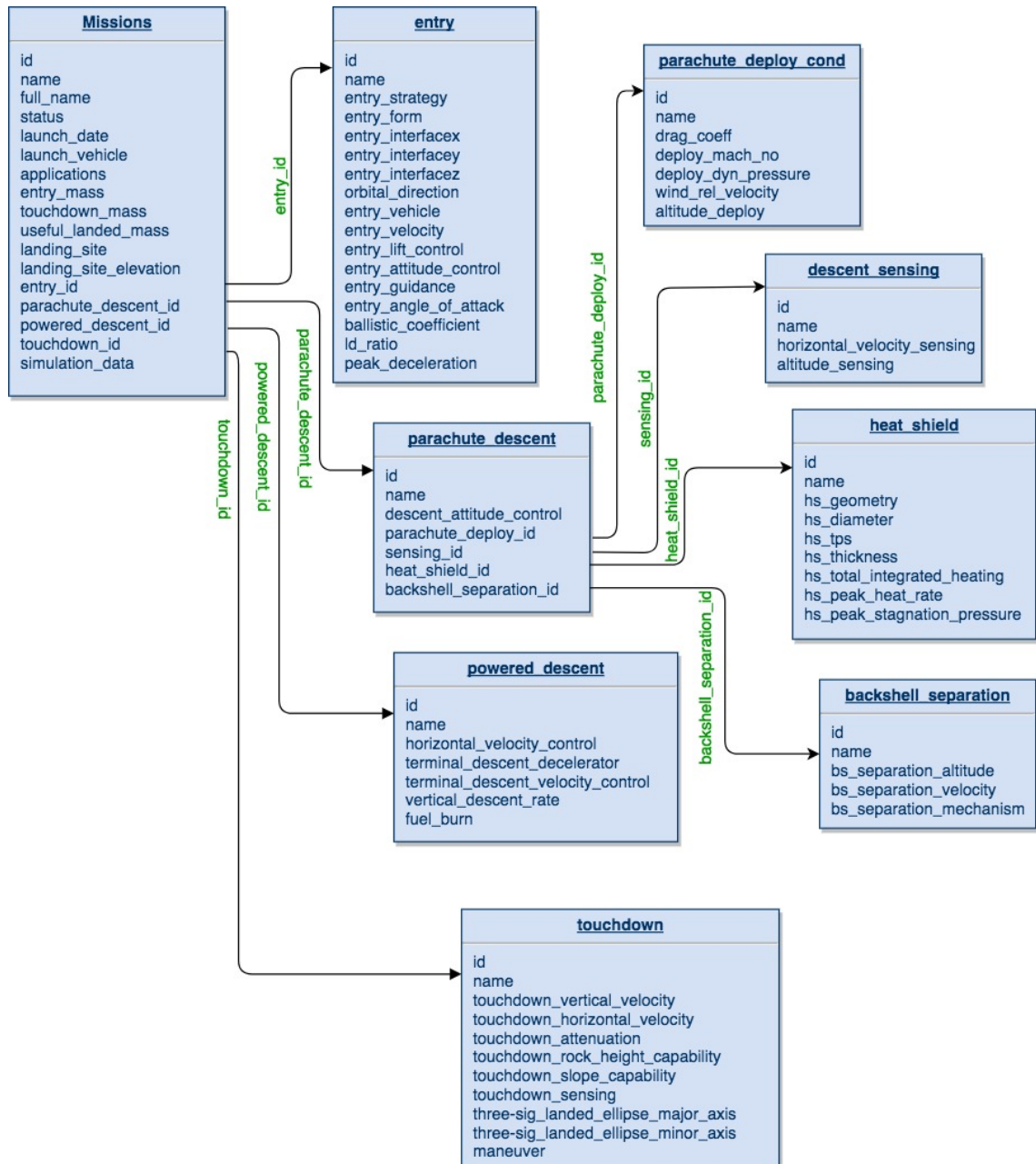


Figure 4.5: EDL database schema.

EDL Scorecard

As stated in Chapter 2, EDL is a highly interdisciplinary problem that results in complex vehicle system designs. Moreover, subsystem analysis and assessing the effect of a particular design becomes and overall performance of an EDL architecture becomes an increasingly complex task. Consequently, the NASA EDL simulation team has created a "scorecard" to provide a common ground to all designers to assess performance for an EDL architecture based on Monte Carlo Simulations for a particular vehicle configuration. The resulting scorecard is a document generated from simulation results and provides a platform for discussion among subsystem designers for dialogue on how subsystem designs affects the overall EDL architecture.

The scorecard provides key EDL metrics and threshold values for critical metrics. Any value that is flagged (e.g. out of spec, unusual) has to be discussed among the team to re-evaluate the design. This document was used as the baseline source of knowledge for the EDL skill knowledge base given that it provides a standardized knowledge that is shared among all EDL groups. A basic representation of a scorecard used to assess the performance of the EDL pre-flight simulation predictions based on reconstructed flight data from MSL is available in Reference [4].

The following chapter is devoted to presenting the preliminary capabilities developed for the Daphne/EDL intelligent assistant in the context of a case study. The goal is to present various plausible use cases in which the assistant can be incorporated into the architecture analysis process. In the first use case, a POST-II simulation was conducted to take a glimpse into the effect of uncertainty in entry velocity on atmospheric skip-out. Daphne is used to ac-

cess the simulation dataset via queries. In the second use-case the user can ask about statistics of a metric and can request historical data on other EDL architectures for comparison. In the third use-case, the user can automatically generate a scorecard for the dataset in study to assess preliminary performance of the proposed architecture.

CHAPTER 5

CASE STUDY

5.1 Use Case 1: Rapid Assessment of Mission-Specific System Performance

As shown in Chapter 3 in Figure 3.3, the EDL team analyzes the outputs after a simulation run to identify any metrics that do not satisfy the requirement values and examine any unusual values. To perform such task, the EDL team generates a scorecard for that particular simulation and examines all of the metrics of interest with the goal of identifying those that are out of spec. For missions such as, MSL, or Mars 2020 have made use of the EDL scorecard as a means for rapid assessment of system performance.

At the moment, Daphne has automated the scorecard generation task in the EDL validation and verification process. For missions with a scorecard template available, such as MSL or Mars 2020, Daphne can be used to automatically generate a scorecard for a particular EDL simulation case. The user can request the values of a particular metric using natural language. Figure 5.1 shows the commands and queries the user can make to obtain such information. First, the user can command Daphne to generate a scorecard for a specific mat-file. Using the scorecard metric threshold values, Daphne can indicate the user or team what metrics are flagged and which metrics are out of spec.

The automatic generation of a scorecard facilitates the task of loading a simulation file and executing the system calls for parsing the scorecard, and searching through the hundreds of metrics for each EDL segment in the architecture.

User: For <i>M2020</i> , load simulation dataset <i>matout.mat</i> ?
Daphne: File loaded
User: For Mars 2020, generate a scorecard
Daphne: Scorecard loaded and populated
User: What metrics are flagged?
Daphne: The flagged metrics in this scorecard are: <ul style="list-style-type: none"> • Touchdown Vertical Velocity 0.82455715102824 < 0.82 m/s is not satisfied • Touchdown Horizontal Velocity 0.3155903077545053 < 0.3 m/s is not satisfied • <=2 bank reversals started 17.86026746656668 <= 0 % is not satisfied • Parachute Inflation Loads 62.3899901065164 < 60 1000 lbf is not satisfied • Maximum Drag Curve Method 62.3899901065164 < 60 kips is not satisfied • RSS X-Y rates at BSS 30.41057788522509 < 30 deg/s is not satisfied
User: What metrics are out of spec?
Daphne: The out of spec metrics in this scorecard are: <ul style="list-style-type: none"> • Parachute Deploy Range Error 8.492602040298564 < 4 km is not satisfied

Figure 5.1: Follow-up questions for Daphne (Historical Database)

However, the knowledge contained in the scorecard can be exploited to enhance Daphne’s capabilities. Other than being the source of expert-knowledge of the key metrics in an EDL architecture, each scorecard contains information on how these values are obtained (e.g. equations and descriptions) that can be used alongside with the simulation data. By running a MATLAB engine in the background, the user can be able to ask for a calculation of a metric from the simulation file and obtain the result in the front-end. Furthermore, thresholds and criteria available in the Scorecard can be used as a form of “rules” for EDL performance assessment.

5.2 Use Case 2: Accessing and Analyzing Simulation Dataset

5.2.1 Context and Goals of the Simulation

As mentioned in Section 2.2 and in Ref [1], high-scientific return missions to Mars will require capabilities to land heavier payloads on the surface of Mars. To overcome the challenges of landing high- β entry vehicles, future missions will require advances beyond the Viking EDL technologies. Whereas robotic missions can make use of lower-mass TPS material and hypersonic decelerating technologies, future manned missions will inevitably require landing capabilities for entry vehicles with ballistic coefficients 3-4 times higher than any mission that has landed on Mars.

Historically, steep EFPA has been preferred given that total heat load on the TPS of the entry vehicle is lower than shallow EFPA for high- β vehicles [33]. However, peak deceleration and heat rate increase. Consequently, as payload mass increases, EDL has shifted to the paradigm of shallower EFPA to reduce altitude of peak deceleration and heat rate. However, decreasing the magnitude of EFPA introduces the likelihood of the entry vehicle skipping out of the atmosphere.

MPF was the first successful ballistic mission landed on the surface of Mars by the United States. This mission demonstrated a low-cost alternative for placing a payload safely on Mars. Success for MPF was dependent on entering the Martian atmosphere at a shallow flight-path angle to increase the altitude at which peak deceleration occurs, increasing timeline for subsequent events in the EDL sequence. For MPF, the skipout angle for the Martian atmosphere was

-11.2 degrees. However, to counterbalance for other uncertainties, the entry trajectory was designed so that the EFPA 3σ was at least 2-degrees steeper than the estimated skipout angle [34].

For the sake of demonstrating Daphne’s current capabilities, this case study takes a glimpse at a high ballistic coefficient vehicle concept using the MSL configuration and the effect of entry velocity in skip-out angle. In reality, more uncertainties are accounted for in such simulation (e.g. atmospheric density, gravity) but in this case study we vary one metric as it would be done in a one-at-a-time sensitivity analysis. At the moment, this task is done in this manner due to the human limitations in the analysis. In the future, we want Daphne to identify these sensitivities to the user without having to use the one-at-a-time approach. The analysis capabilities for EDL in Daphne provides analysis capabilities to the user, extracting useful information on EDL metrics from POST-II simulation results. The user can request statistical data on a given parameter. The user can also ask for nominal values from previous missions to compare EDL performance metrics. For missions with established system-level requirements, Daphne can automate the process of generating a scorecard and respond to the user indicating metric values and which set of metrics is flagged or out-of-spec.

5.2.2 Simulation Setup

For this study, a 3-degree-of-freedom trajectory analysis was conducted using POST-II-based Monte Carlo simulation for a ballistic entry vehicle. The simulation incorporated the planet model and the revised planetary constants for Mars

as well as the atmospheric database available in POST-II. The gravity model employed assumes a spherical planet of radius R with gravitational constant μ .

The following subsections describe the parameters and assumptions for the trajectory analysis of a high ballistic coefficient entry vehicle assuming nominal atmospheric density at the entry interface and a trajectory analysis for varying atmospheric density conditions. For all cases the same 70-deg sphere-cone heat shield and TPS were used. The same entry states were applied to all simulation cases. The parameters and values are shown in Table 5.2.2.

Table 5.1: Entry States and Heat Shield Parameter for Simulation

Parameter	Nominal
Heat shield diameter(m)	4.0
Heat shield material	SLA-561
Drag coefficient	1.47
Initial azimuth (deg)	30.0
Initial declination angle (deg)	-4.5
Geocentric Radius (km)	3522.2

5.2.3 Skipout Angle Due to Dispersion Entry Velocity and EFPA

Entry velocity into a planetary atmosphere is dependent on launch date and events leading up to cruise stage separation. Nevertheless, the first part of the case study takes a glimpse at the effect of uncertainty in the entry velocity in the

Table 5.2: Cases evaluated for Monte Carlo simulation to measure the Effect of uncertainty in entry velocity and EFPA to skipout angle

Case	1	2	3	4	5	6	3σ
Ballistic Coefficient (kg/m^2)	200	220	240	260	280	300	-
Entry flight-path angle (deg)	-12	-12	-12	-12	-12	-12	5
Entry Velocity (km/s)	5	5	5	5	5	5	5

range of 5.0 - 6.0 km/s in skip-out angle for ballistic coefficients of 200, 220, 240, 260, and 300 kg/m^2 . Given that the case study consists of a high-level assessment of uncertainty in skipout angle, large dispersions of the EFPA and entry velocity were used to capture a wider range of behavior in entry conditions for different ballistic coefficient entries. The morphological box depicted in Table ?? presents the cases examined with the respective standard deviations for the Monte Carlo trajectory analysis.

5.2.4 Results

Sensitivity of EFPA for Skip-out due to Variations in Entry Velocity

The six individual Monte Carlo simulation presented in Table 5.2 were executed with with 8001 cases being evaluated in each simulation. Figure 5.2 shows the variation of skipout angle due to uncertainty for the entry velocity as the ballistic coefficient of the entry system increased. As expected, as the ballistic coefficient of the vehicle increases, the tendency will be to skipout of the atmosphere at steeper entry flight path angles. In addition, Figure 5.2 also shows that entry vehicles with high kinetic energy, and thus high entry velocities, have a higher

probability of skipping out of the atmosphere.

In reality, there many sources of variability in entry parameters during atmospheric entry. However, given that the simulation conducted is a very simplified 3-DOF ballistic entry case with no lift or guidance, the number of uncertain variables that can be dispersed is limited. Consequently, for the sake of demonstrating the sensitivity of skipout angle to entry velocity and EFPA, two critical EDL entry conditions, these parameters were selected as the primary sources of uncertainty. A high 3-sigma dispersion analysis was used to attempt to capture any potential uncertainties that affect entry velocity and EFPA.

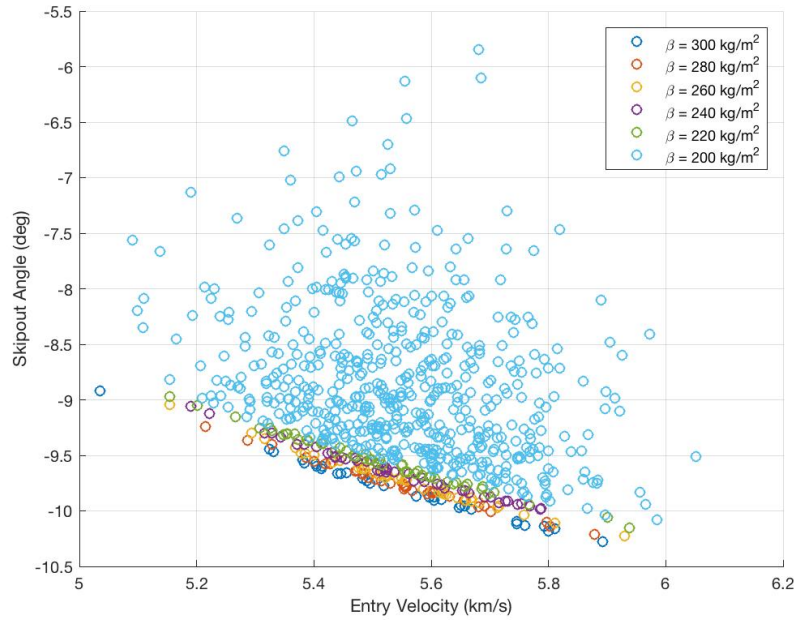


Figure 5.2: Skipout angle variation due to uncertainty in entry velocity and EFPA

Figure 5.3 shows the steepest entry flight-path angles (25%-tile) of the cases shown in Figure 5.2. The range of skipout angles for each ballistic coefficient evaluated is below the 1.0 degree range. The values for each case evaluated

present the threshold of values for a particular vehicle configuration as a function of ballistic coefficient and entry velocity. In order to minimize the risk of skipout, nominal values for entry vehicle system design must fall below the boundaries formed by each vehicle configuration. In other words, unless ballistic coefficient is reduced, entry velocities need to be reduced to avoid the risk of entering the skipout region. Similarly, low β vehicles at high entry velocities undergo the same risk of skipout at steep angles.

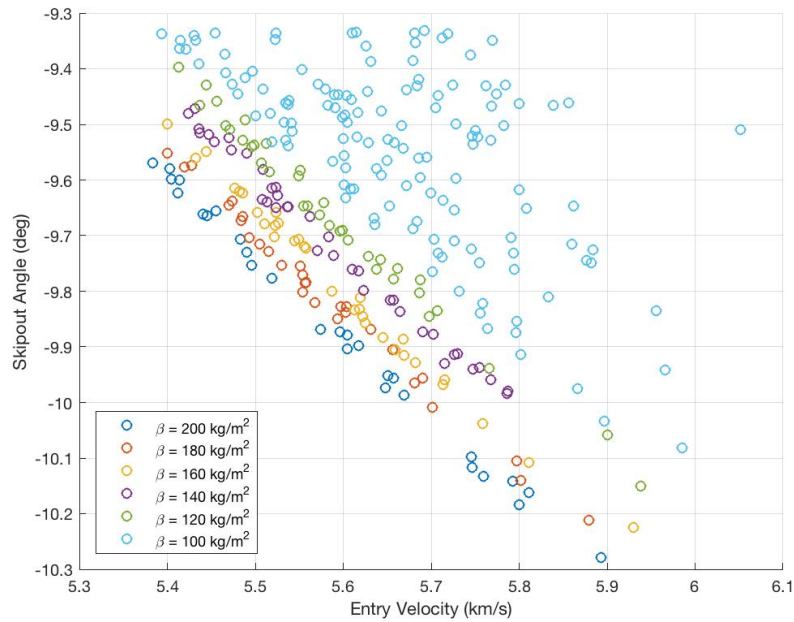


Figure 5.3: Steepest Skipout angle variation due to entry velocity

5.2.5 Interacting With Daphne

Typically after executing a POST II-based Monte Carlo Simulation, EDL system validation and verification requires that simulators examine test results with the goal of identifying any discrepancies in the results, as shown in Chapter 3,

Figure 3.3. Any anomaly identified is discussed between model developers and simulators to resolve potential issues in the code implementation or in the EDL architecture itself.

As in the first use case, the EDL design team has to locate the simulation file of interest and load it for analysis. If more than one simulation case is to be examined, the user has to perform additional manual labor to repeat the same procedure for all datasets. Although this task may seem at first a non-trivial task to automate given that the user can simply load the data set, and evaluate statistical functions, in reality this question often arises during discussions in EDL teams. And often, individual members of the team have to search for datasets, load them, and examine them during design discussions, making the discussion flow very inconsistent. For example, for examining the simulations for potential landing site locations the team members may ask “What is the peak deceleration for landing site A” followed by “What was the peak deceleration for landing sites B and C?” With Daphne, we want to automate the process of loading these datasets via commands and have the user ask about the statistics of a given metric. If the user is interested in looking into another dataset, she/he can simply ask Daphne to load a different dataset. We envision that EDL teams can make use of this intelligent assistant as an additional team member that collects information for the team and communicates it back to the team in a human-like manner.

In the simulation executed for this study, once results are obtained, the user can add her or his simulation data into the repository of a particular mission. The user can ask Daphne for a list of simulations available (if needed) and can proceed to command Daphne to load the simulation file that is to be studied.

Figure 5.4 shows a sample dialogue between the user and Daphne for extracting information on the simulations conducted to study the effect of dispersion in entry velocity for an entry vehicle with ballistic coefficient of 300 kg/m^2 .

User: For case study , load simulation dataset <i>skipout_cases_B300.mat</i> ?
Daphne: File loaded
User: What are the <i>statistics</i> for <i>relative flight path angle entry interface</i> ?
Daphne: For relative flight path angle at entry interface in skipout_cases_B200.mat, the statistics are: <ul style="list-style-type: none"> • max = -9.569803212074994 • min = -10.279281672469995 • mean = -9.786955234902459 • variance = 0.025098226305201227 • std = 0.15842419734750504 • 3s = -9.311682642859944 • mean = -9.786955234902459 • -3s = -10.262227826944974 • 0.13% = -10.266234160137532 • 1.00% = -10.213899811204497 • 10.00% = -9.985595841814497 • 50.00% = -9.753010773615998 • 90.00% = -9.5729120745931 • 99.87 = -9.569862194617745 • high 99.89 - median = 0.183148578998253 • high 99.87 - mean = 0.183148578998253 • median - low 99.87 = 0.5132233865215348 • mean - low 99.87 = 0.4792789252350733

Figure 5.4: Use Case

When the user asks Daphne to load the data set, the question is classified as an EDL-type of command and it executes an executable function that loads the data set requested. Daphne responds indicating that the file has been loaded successfully. Once the dataset of interest is identified and loaded by Daphne, the user can ask for statistics of a specific output variable from the Monte Carlo simulations using the variable name (if the user happens to know it) or using a natural language description of the variable. In Figure 5.4 the user asks Daphne on the statistics for entry flight path angle at the entry interface. At the moment, Daphne only provides text description of the metrics. One of the desired capabilities is for Daphne to display graphics of the probability distributions of the results. This task is currently under development for the EDL assistant inter-

face. Through web-based interface way the user can visualize the distributions and obtain information on the results by hovering or clicking on the figure.

<p>User: What are the statistics for <i>relative velocity entry interface</i>?</p>
<p>Daphne:</p> <p>For relative velocity at entry interface in skipout_cases_B300.mat, the statistics are:</p> <ul style="list-style-type: none"> • max = 5985.258260269001 • min = 5383.634601101001 • mean = 5646.996929961758 • variance = 16346.496654117065 • study = 127.8534186250687 • 3s = 6030.557185836964 • mean = 5646.996929961758 • -3s = 5263.436674086552 • 0.13% = 5388.248513419503 • 1.00% = 5404.0449474709 • 10.00% = 5491.683695637999 • 50.00% = 5636.9268354095 • 90.00% = 5957.2778363834 • 99.87 = 5980.6373197401135 • high 99.89 - median = 343.7104843306133 • high 99.87 - mean = 343.7104843306133 • median - low 99.87 = 248.67832198999713 • mean - low 99.87 = 258.74841654225474

Figure 5.5: Follow-up questions for Daphne

Figure 5.5 presents a follow up question the user makes. Considering Daphne can recall what dataset is in the working memory, the user can ask for other variable information without having to tell Daphne to reload the data set. The user can continue to ask for statistics of variables of interest in a data set, thus reducing the cognitive load required by the user. In this use-case the user requests velocity at the entry interface, a critical parameter at the entry interface of the planet. Although this question is not depicted in the figure presented, a logical follow up question would be "what are the statistics for peak deceleration" given that this parameter is dependent on EFPA and entry velocity.

If the user is interested in a distinct data set for a particular vehicle configuration, he or she simply has to ask Daphne to load the data set. If the user continues to follow this method, a table such as the one depicted in Table 5.3 can be constructed, automating the process of having to manually search, reload

Table 5.3: Simulation results for skipout angle due to sensitivities to entry velocity and ballistic coefficient.

	Ballistic Coefficient β (kg/m^2)											
	200		220		240		260		280		300	
Statistic	γ_{EI} (deg)	V_{EI} (km/s)	γ_{EI} (deg)	V_{EI} (km/s)	γ_{EI} (deg)	V_{EI} (km/s)	γ_{EI} (deg)	V_{EI} (km/s)	γ_{EI} (deg)	V_{EI} (km/s)	γ_{EI} (deg)	V_{EI} (km/s)
Min	-10.0816	5.3937	-10.150	5.4131	-10.150	5.4237	-10.225	5.4004	-10.225	5.3999	-10.2793	5.3836
Mean	-9.5478	5.6670	-9.6210	5.6620	-9.6673	5.6575	-9.7257	6.6531	-9.7647	5.6461	-9.787	5.6469
Max	-9.3320	6.0519	-9.4304	6.0519	-9.4805	6.0519	-9.5232	6.0519	-9.5562	5.9853	-9.5698	5.9852
1σ	0.1634	0.1293	0.1592	0.1283	0.1560	0.1286	0.1562	0.1277	0.1527	0.1236	0.1589	0.1281
3σ	0.4906	0.3880	0.4777	0.3851	0.4680	0.3860	0.4686	0.3834	0.4680	0.3710	0.4766	0.3845

and configure datasets. This way the user, or the design team can quickly extract data for examining all cases of interest in a single item. Table 5.3 presents a summary of the results obtained in the simulation concerning entry velocity and skipout angle for all six ballistic coefficient vehicles evaluated. For the steepest flight path angles, the percent increase in EFPA skipout angle from a 200 to a 300 kg/m^2 is roughly 2.5%. For lower β entry vehicles, skipout angle is still possible at high entry velocities and skipout angle can be just as steep as high β entry with lower entry velocity.

Although Daphne automates the task of loading each individual dataset manually and computing the desired statistics, Daphne can be made more helpful to the team by loading multiple datasets and providing the user a summary of the parameter requested for all simulations considered.

Figure 5.6 shows the queries made to Daphne to obtain peak deceleration values for cases for an entry vehicle of ballistic coefficient of 300 kg/m^2 . First, the user requests to load the data set of landed cases, then the user asks for the magnitude of the peak deceleration. Following this question, the user can proceed to ask for the landed cases in which lofting occurred. Often the EDL

```
User: For the case study, load simulation dataset landed_cases_B300.mat ?
Daphne: File loaded
User: What are the statistics for g load peak deceleration?
Daphne: For g load peak deceleration in landed_cases_B200.mat, the statistics are:
• max = 13.93247868028832
• min = 1.4667277367822444
• mean = 5.831378403063296
• variance = 5.102396052235452
• std = 2.258848390714935
• 3s = 12.607923575208101
• mean = 5.831378403063296
• -3s = -0.94516676908151
• 0.13% = 1.5226761063646483
• 1.00% = 1.6297617538374407
• 10.00% = 2.6808082506976527
• 50.00% = 5.8199595400260655
• 90.00% = 10.931425158584329
• 99.87 = 12.325237302606595
• high 99.89 - median = 6.50527776258053
• high 99.87 - mean = 6.50527776258053
• median - low 99.87 = 4.297283433661417
• mean - low 99.87 = 4.308702296698647
```

Figure 5.6: Follow-up questions for Daphne

team is interested in lofting considering that up to some extent, it helps minimize the magnitude of the peak deceleration. However, lofting increases the probability of skip-out. Figure 5.7 shows a plot of peak deceleration for three randomly selected cases with lofting. Figure 5.8 shows the trajectories for these same cases.

To make Daphne more helpful to the EDL team we want for Daphne to be able to identify stressing cases, such as lofting or atmospheric skipout, for example. Achieving this task is one of the next steps for Daphne.

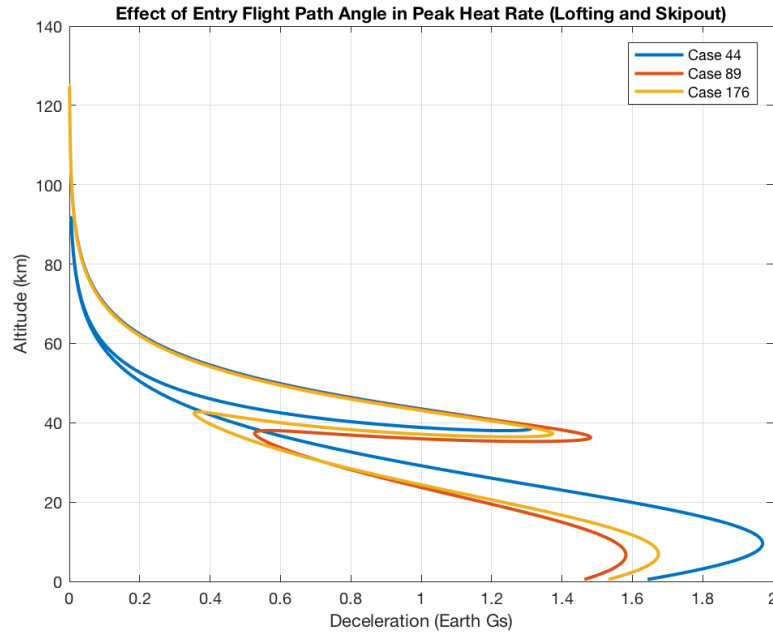


Figure 5.7: Deceleration for three random cases in which lofting occurs for an entry vehicle of $\beta = 300\text{kg/m}^2$

5.3 Use Case 3: Comparing Simulation Results to Nominal Values of Previous Landed Missions

Considering the inherent similarity due to technology heritage for all successful missions to Mars made by the U.S., assessment of EDL metric performance often relies on analysis by similarity. Because of this practice, the EDL historian was incorporated into the skill to rapidly provide the users, or EDL team, relevant information on a particular EDL architecture. All of the information is available in one source and minimizes the need for the user at the front end to have to search from multiple sources of information.

For example, in the second use case study we examined sensitivities due to ballistic coefficient, entry velocity, and entry flight path angle for a ballistic en-

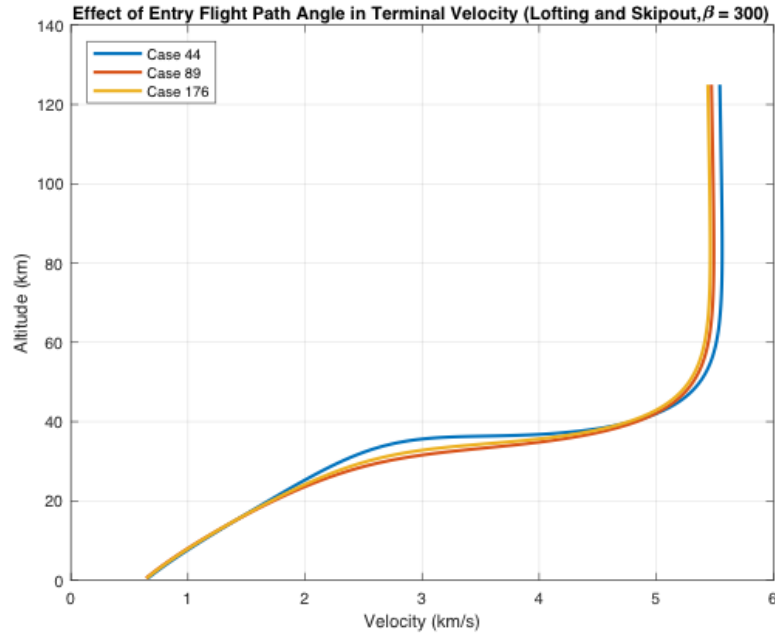


Figure 5.8: Velocity as a function of altitude for three random cases in which lofting occurs for an entry vehicle of $\beta = 300 \text{ kg/m}^2$

try mission. Once the user obtains simulation results, she or he can ask Daphne about these parameters for previous missions. For example, Figure 5.9 shows the dialogue we would have with Daphne after examining our results. First, we assume that the user is interested in looking into entry flight path angles for previous ballistic missions to Mars. In the first query, the user asks Daphne *"What is the ballistic coefficient for MPF?"*. Daphne determines that this statement is a database query, and extracts the features of interest from the SQL database. By making these queries for all relevant parameters corresponding to MPF, Daphne provides multiple responses indicating the ballistic coefficient was 63 kg/m^2 and has an entry flight path angle was -14.06 degrees and the peak deceleration was 20 G. We can see that a higher peak deceleration is attributed to a high entry velocity (7.26 km/s), and peak deceleration is much higher than for any other previous missions.

Looking further into EDL metrics for missions with high ballistic coefficient we have MSL, the largest payload successfully landed on Mars up to the present day. Using Daphne, the user simply has to request the parameter and mission, no additional effort is required from the user and she or he can easily switch between query types. We can see that when the user asks for the ballistic coefficient of MSL, the ballistic coefficient is over two times larger than the ballistic coefficient for MPF. Furthermore, the EFPA for MSL was steeper (-15 deg). However, peak deceleration was 8 Gs lower, likely due to the fact that MSL used a lifting-guided entry and due to a higher mass system, entry velocity was lower. Although Daphne does not provide these rationales yet, experts in the EDL can obtain the nominal values for previous missions and assess differences in performance using a single source of knowledge.

At the moment the historian only provides the nominal value for a particular mission. However, the Historical Database can be exploited to provide critique-type comments to the user for evaluating EDL architectures. Combined with the rules available in the expert knowledge base, quality of critiques can be enhanced by making use of data from previous missions.

User: For MPF , what is the ballistic coefficient ?
Daphne: For mission MPF the entry ballistic coefficient is 63.0 kg/m^2
User: For MPF , what is the entry flight path angle ?
Daphne: For mission MPF the entry flight-path angle is -14.06 deg
User: For MPF , what is the peak deceleration ?
Daphne: For mission MPF the entry peak deceleration is 20.0 G
User: For MPF , what is the entry velocity ?
Daphne: For mission MPF the entry velocity is 7.26 km/s
User: For MSL , what is the ballistic coefficient ?
Daphne: For mission MPF the entry ballistic coefficient is 140 kg/m^2
User: For MSL , what is the entry flight path angle ?
Daphne: For mission MPF the entry flight-path angle is -15.02 deg
User: For MSL , what is the peak deceleration ?
Daphne: For mission MPF the entry peak deceleration is 12.0 G
User: For MSL , what is the entry velocity ?
Daphne: For mission MPF the entry velocity is 5.84 km/s

Figure 5.9: Follow-up questions for Daphne (Historical Database)

CHAPTER 6

CONCLUSIONS

In this thesis we introduced a cognitive assistant for EDL based on Daphne, who currently specializes in providing support to architect constellations of satellites. The main objective is to use Daphne/EDL assistant as a platform for Intelligent Data Understanding technologies for performance analysis, specifically for entry descent and landing architecture analysis. Advancing Daphne's capabilities serve as an aid to EDL design teams to rapidly evaluate EDL performance metrics, identify high information content data and improve the uncertainty characteristics of inferences made by informed decisions using multiple sources of knowledge. Daphne can take off cognitive load from EDL architecture analysts by providing basic analysis and background information on metrics of interest in EDL architectures. Furthermore, Daphne automates the steps required to obtain such information by making queries to the database, loading data set for the user with the necessary functions and makes basic calculations. At the moment, Daphne can provide basic statistics of metrics in a simulation data set and can generate scorecards for missions by means of natural language. The case study presented in this thesis illustrates three use cases on how an expert or EDL team can extract information from Daphne by means of the historical data base or the scorecard to obtain insights on metrics of interest.

While having already some capabilities, the current implementation of Daphne/EDL assistant is in its early stages. First, Daphne's knowledge on variable and metric descriptions is limited. For example, the user can ask for peak deceleration, which Daphne knows. However, if the user uses different descriptions such as "g-load", Daphne does not recognize that these metrics are the

same. Improvements have to be made with natural language processing for EDL metrics. Second, at the moment identifying cases in an EDL data set is still up to the user. For example, in this case study, the user had to identify the cases in which lofting occurred by inspecting the output data set. As future work, Daphne shall identify any stress cases such as *"In what cases does lofting occur?,"* *"In what percent of cases does lofting occur?,"* *"In what cases is there skip-out from the atmosphere?,"* *"In what percent of cases does skip-out occur?,"* among other critical events of interest to the EDL design team.

Another critical limitation is assessing sensitivity of input metrics of the architecture. At the moment the sensitivity analysis conducted on the data set was employed using the current technique the EDL design team uses: "one-at-a-time" sensitivity analysis. Although this approach is straightforward, it fails to identify dependencies and interactions between input variables. In the near future, we want Daphne to conduct sensitivity analysis on EDL data sets such as: variance-based sensitivity, such as first order and total order sensitivity indexes; or density-based sensitivity analysis. Finally, one limitation of the existing IDU technologies fail to capture expert knowledge in the loop. Mixed-initiative approaches for IDU technologies in which Daphne's EDL skill provides feedback on an EDL architecture design without having the user request such information are also under consideration for future work [30] [35].

BIBLIOGRAPHY

- [1] Braun, M., Manning, R. (2007) Mars Exploration Entry, Descent, and Landing Challenge, *Journal of Spacecraft and Rockets*, 44(2), pp. 310-323.
- [2] Wolf, A. A. et al. (2011) Toward Improved Landing Precision on Mars, in *IEEE Aerospace Conference Proceedings*. doi: 10.1109/AERO.2011.5747243.
- [3] Mitcheltree, R. et al. (2006) Mars Science Laboratory Entry, Descent and Landing System Verification and Validation Program, 2006 IEEE Aerospace Conference, pp. 1-6. doi: 10.1109/AERO.2006.1655799.
- [4] Striepe, S. A. et al. (2006) Mars Science Laboratory Simulations for Entry, Descent, and Landing, *Journal of Spacecraft and Rockets*. doi: 10.2514/1.19649..
- [5] Lockwood, Mary K. et al. (2006) Entry Configurations and Performance Comparisons for the Mars Smart Lander, *Journal of spacecraft and rockets*, 43(2), p. 258. doi:10.2514/6.2002-4407.
- [6] Way, D. W., Davis, J. L. and Shidner, J. D. (2013) Assessment of the Mars Science Laboratory Entry, Descent, and Landing Simulation, in *Advances in the Astronautical Sciences*, pp. 563-581. doi: 10.1109/AERO.2013.6497404.
- [7] Cruz, J. R. et al. (2014). Reconstruction of the Mars Science Laboratory Parachute Performance, *Journal of Spacecraft and Rockets*. doi: 10.2514/1.A32816.
- [8] J. Allen, H & J. Jr. Eggers, A. (1957). A Study of the Motion and Aerodynamic Heating of Missiles Entering the Earth's Atmosphere at High Supersonic Speed.
- [9] Griffin, M. D. and French, J. R. (2004) *Space Vehicle Design*, AIAA Education Series.
- [10] Regan, F. J. and Anandakrishnan, S. M. (1993) *Dynamics of Atmospheric Re-Entry*. American Institute of Aeronautics & Astronautics (AIAA Education Series).
- [11] Wiesel, W. E. (1989) *Spaceflight Dynamics*. McGraw-Hill (McGraw-Hill Series in Aeronautical and Aerospace Engineering).

- [12] Putnam, Z. R. and Braun, R. D. (2015). Extension and Enhancement of the Allen-Eggers Analytical Ballistic Entry Trajectory Solutions, *Journal of Guidance, Control, and Dynamics*. doi: 10.2514/1.G000846.
- [13] Platus, D. H. Ballistic Re-entry Vehicle Flight Dynamics. *Journal of Guidance and Control*, 5 (1982), 416.
- [14] Subrahmanyam, P. (2009). High-Fidelity Aerothermal Engineering Analysis for Planetary Probes Using DOTNET Framework and OLAP Cubes Database, *International Journal of Aerospace Engineering*. doi: 10.1155/2009/326102.
- [15] Hughes, S. et al. (2011). Hypersonic Inflatable Aerodynamic Decelerator (HIAD) Technology Development Overview, in 21st AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar. doi: 10.2514/6.2011-2524.
- [16] Clark, I. G. et al. (2008). Supersonic inflatable aerodynamic decelerators for use on future robotic missions to Mars, in *IEEE Aerospace Conference Proceedings*. doi: 10.1109/AERO.2008.4526289.
- [17] Cruse, T. A. (1997). *Reliability-based mechanical design*. New York: M. Dekker.
- [18] Kroese, D. P. et al. (2014) Why the Monte Carlo Method is so Important Today, *Wiley Interdisciplinary Reviews: Computational Statistics*. doi: 10.1002/wics.1314.
- [19] Hamby, D. M. (1994) A Review of Techniques for Parameter Sensitivity Analysis of Environmental Models, *Environmental Monitoring and Assessment*. doi: 10.1007/BF00547132.
- [20] Petrick, D. et al. (2014). SpaceCube v2.0 Space Flight Hybrid Reconfigurable Data Processing System, in *IEEE Aerospace Conference Proceedings*. doi: 10.1109/AERO.2014.6836226.
- [21] Chien, S. et al. (2005) An Autonomous Earth-Observing Sensorweb, *IEEE Intelligent Systems*. doi: 10.1109/MIS.2005.40.
- [22] NASA (2015) NASA Technology Roadmaps TA 11: Modeling, Simulation, Information Technology, and Processing, *NASA Technology Roadmaps*.

- [23] Ghahramani, Z. (2015) Probabilistic Machine Learning and Artificial Intelligence, *Nature*. doi: 10.1038/nature14541.
- [24] Schum, D. A. et al. (2014) Toward Cognitive Assistants for Complex Decision Making Under Uncertainty, *Intelligent Decision Technologies*, 8(3), pp. 231250.
- [25] zyurt, E. and Dring, B. (2012) A Cognitive Assistant for Supporting Air Target Identification on Navy Ships, *IFAC Proceedings Volumes*. doi: 10.3182/20120215-3-AT-3016.00082.
- [26] Lunenburg, Fred C. The Decision Making Process, *National Forum of Educational Administration Supervision Journal*. Vol. 27. No. 4. 2010.
- [27] Tokadli, G. & Dorneich, M.C., (2017). Development of Functionality Matrix for a Cognitive Assistant on Long Distance Space Missions, *Proceedings of Human Factors and Ergonomics Society*, October 9-13, Austin-Texas.
- [28] Hyunseung Bang, Antoni Viros Martin, Arnau Prat, and Daniel Selva. "Daphne: An Intelligent Assistant for Architecting Earth Observing Satellite Systems", 2018 AIAA Information Systems-AIAA Infotech at Aerospace, AIAA SciTech Forum, (AIAA 2018-1366) <https://doi.org/10.2514/6.2018-1366>.
- [29] Sellers, P. H., The Theory and Computation of Evolutionary Distances: Pattern Recognition, *Journal of Algorithms*, Vol. 1, No. 4, 1980, pp. 359 373.
- [30] Bang, H. & Selva, D. (2016). iFEED: Interactive Feature Extraction for Engineering Design, In *Proceedings of the ASME 2016 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE*.
- [31] Viros, A. (2017). Design and Development of a Cognitive Assistant for the Architecting of Earth Observing Satellites (Bachelor thesis). Universitat Politcnica de Catalunya (UPC) BarcelonaTech, Spain.
- [32] Mall, Kshitij & J. Grant, Michael. (2016). High Mass Mars Exploration using Slender Entry Vehicles. 10.2514/6.2016-0019.
- [33] Dutta, S. et al. (2012). Mission Sizing and Trade Studies for Low Ballistic Coefficient Entry Systems to Venus, in *IEEE Aerospace Conference Proceedings*. doi: 10.1109/AERO.2012.6187002.

- [34] Spencer, D. A. and Braun, R. D. (1996). Mars Pathfinder atmospheric entry - Trajectory design and dispersion analysis, *Journal of Spacecraft and Rockets*. doi: 10.2514/3.26819.
- [35] Allen, J., Guinn, C. I., and Horvtz, E., Mixed-initiative interaction, *IEEE Intelligent Systems and their Applications*, Vol. 14, No. 5, 1999, pp. 1423.